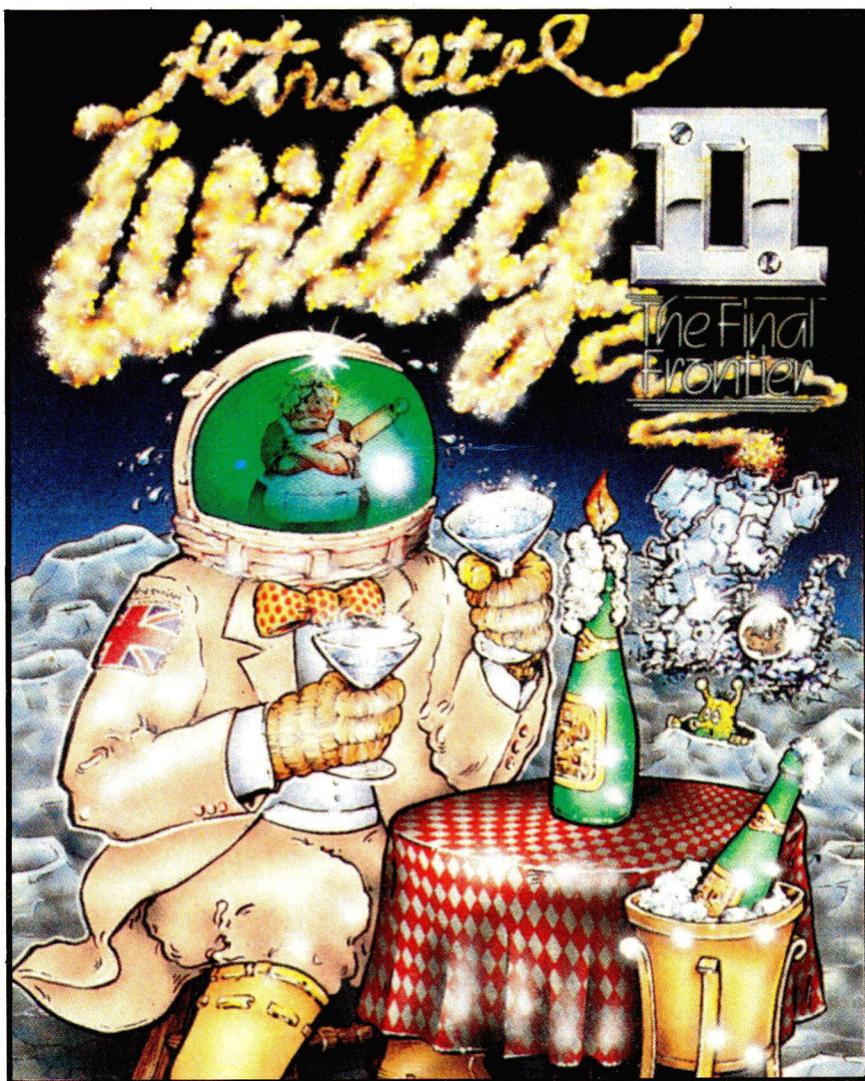
LA SOFTRIVISTA JACKSON PER GLI UTENTI MSX

JACKSON SOFT compilation MSX

SUPERGAME:

JET SET WILLY II pag. 4

GUIDA ALL'INPUT pag. 9

LISTATI:

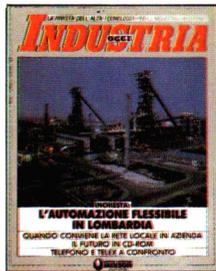
AGGUATO SUL FONDO pag. 10

DISASSEMBLER pag. 14

ARTICOLI:

CORSO PRATICO DI LINGUAGGIO MACCHINA pag. 18

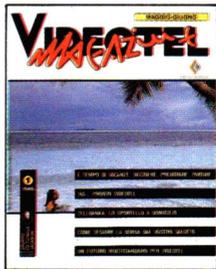
PICCOLI TRUCCHI pag. 19

Spedizione in abbonamento postale - Gruppo III - 70

INDUSTRIA OGGI

Il mensile dell'alta tecnologia nell'industria moderna: soluzioni applicative e nuovi orientamenti in R&S, produzione e servizi.
Abb. annuo
10 numeri
lire 41.000
anziché lire ~~50.000~~

ELETTRONICA OGGI

La più autorevole rivista italiana di componenti, strumentazione ed elettronica professionale.
Abb. annuo
20 numeri
lire 64.000
anziché lire ~~80.000~~

VIDEOTEL MAGAZINE

La rivista dei nuovi servizi interattivi telematici: applicazioni, fornitori di informazione, utilizzo.
Abb. annuo
6 numeri
lire 20.000
anziché lire ~~24.000~~

INFORMATICA OGGI

L'informatica professionale: dall'elaborazione dati all'office automation. Servizi speciali e anticipazioni esclusive dalla Silicon Valley.
Abb. annuo
11 numeri
lire 40.000
anziché lire ~~49.500~~

È JACKSO

CAMPAGNA ABBONAMENTI 86-87

GRUPPO EDITO JACKS DIVISIONE PER

Sì desidero abbonarmi alle seguenti riviste Jackson:

Rivista	Numeri	Prezzo
☐ INDUSTRIA OGGI	numeri 10	L. 41.000
☐ ELETTRONICA OGGI	numeri 20	L. 64.000
☐ AUTOMAZIONE OGGI	numeri 11	L. 46.000
☐ ELETTRONICA, STRUM. & AUTOM. PRODUCT NEWS	numeri 11	L. 26.000
☐ TRASMISSIONE DATI E TELECOMUNICAZIONI	numeri 11	L. 36.000
☐ VIDEOTEL MAGAZINE	numeri 6	L. 20.000
☐ INFORMATICA OGGI	numeri 11	L. 40.000
☐ PC & COMPUTER PRODUCT NEWS	numeri 11	L. 27.000
☐ COMPUTER GRAFICA & APPLICAZIONI	numeri 4	L. 19.000
☐ PC WORLD MAGAZINE	numeri 11	L. 44.000
☐ BIT	numeri 11	L. 43.000
☐ SUPERCOMMODORE 64 & 128	numeri 11	L. 66.000
☐ NOI C128 E C64 con cassetta	numeri 11	L. 70.000
☐ NOI C128 e C64 con disco	numeri 11	L. 115.000
☐ OLIVETTI PRODEST USER	numeri 6	L. 15.000
☐ VIDEOGIOCHI & COMPUTER	numeri 11	L. 18.000
☐ COMPUSCUOLA	numeri 9	L. 21.000
☐ ELETTRONICA HOBBY	numeri 12	L. 32.000
☐ STRUMENTI MUSICALI	numeri 11	L. 35.000
☐ NAUTICAL QUARTERLY	numeri 4	L. 70.000
☐ AUTOMOBILE QUARTERLY	numeri 4	L. 69.500

☐ L'abbonamento dovrà decorrere dal mese di ____________

Nome e Cognome ____________

Via e n. ____________ Tel. ( ) ____________

Cap ________ Località ____________ Prov. ______

Per il pagamento ☐ Allego assegno n. ____________ di L. ____________

Banca ____________

☐ Ho effettuato versamento di L. ____________ sul c/c postale n. 11666203 intestato a Gruppo Editoriale Jackson - Milano e allego fotocopia della ricevuta.

☐ Ho effettuato versamento di L. ____________ tramite vaglia postale o telegrafico e allego fotocopia ricevuta.

☐ Vi autorizzo ad addebitare l'importo di L. ____________ sulla carta di credito.

☐ VISA ☐ AMERICAN EXPRESS ☐ DINERS CLUB

N. ____________ Data di scadenza ____________

Data ____________ Firma ____________

ABBONARSI
È FACILE!
UTILIZZATE
QUESTO TAGLIANDO
E SPEDITELO,
IN BUSTA CHIUSA, A:

GRUPPO EDITORIALE JACKSON S.p.A.

UFFICIO ABBONAMENTI

VIA ROSELLINI, 12
20124 MILANO

COMPUSCUOLA

L'informatica nella didattica: problemi, esperienze e prospettive del computer nel mondo della scuola.
Abb. annuo
9 numeri
lire 21.000
anziché lire ~~27.000~~

VIDEOGIOCHI NEWS

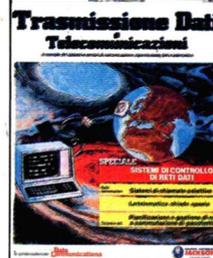
Il giornale di videogames e home computer, con recensioni di giochi sempre nuovi.
Abb. annuo
11 numeri
lire 18.000
anziché lire ~~22.000~~

AUTOMAZIONE OGGI

Robotica, controllo numerico, CAD/CAM, sistemi flessibili... problemi e soluzioni per la nuova automazione industriale.
Abb. annuo
11 numeri
lire 46.000
anziché lire ~~55.000~~

ELETTRONICA, STRUMENTAZIONE & AUTOMAZIONE PRODUCT NEWS

Il giornale di tutti i nuovi prodotti dell'elettronica, della strumentazione dell'automazione industriale e dell'elettronica medicale.
Abb. annuo
11 numeri
lire 26.000
anziché lire ~~33.000~~

TRASMISSIONE DATI E TELECOMUNICAZIONI

Il mensile dei nuovi sistemi di telecomunicazioni, data communications e telematica.
Abb. annuo
11 numeri
lire 36.000
anziché lire ~~44.000~~

PC & COMPUTER PRODUCT NEWS

Il giornale di tutti i nuovi prodotti hardware e software dell'informatica: dai medi sistemi al Personal Computer.
Abb. annuo
11 numeri
lire 27.000
anziché lire ~~33.000~~

PC WORLD MAGAZINE

La prima e unica rivista italiana per gli utenti di personal computer IBM Olivetti e compatibili.
Abb. annuo
11 numeri
lire 44.000
anziché lire ~~55.000~~

BIT

La prima rivista europea, la più famosa e autorevole in Italia, di personal, home, business computer, software e accessori.
Abb. annuo
11 numeri
lire 43.000
anziché lire ~~55.000~~

COMPUTER GRAFICA & APPLICAZIONI

La rivista della grafica e dell'immagine con il computer: applicazioni pratiche per l'industria, le professioni e i servizi. Dal CAD/CAM alla pubblicità.
Abb. annuo
4 numeri
lire 19.000
anziché lire ~~24.000~~

SUPERCOMMODORE 64 & 128

La prima rivista con cassette programmi, dedicate agli utenti home computer Commodore 64 e 128.
Abb. annuo
11 numeri
lire 66.000
anziché lire ~~82.500~~

NOI C128 E C64

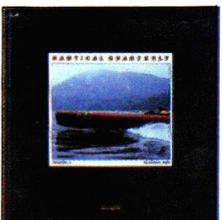
La rivista con disco o cassetta dei package professionali, modelli applicativi e giochi intelligenti.
Abb. annuo 11 numeri
(versione con cassetta)
lire 70.000
anziché lire ~~99.000~~
(versione con disco)
lire 115.000
anziché lire ~~144.000~~

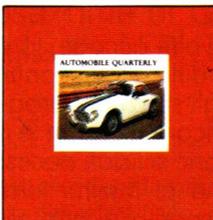
OLIVETTI PRODEST USER

L'unica rivista per gli utenti dei sistemi Olivetti Prodest PC128 e PC128S. Una guida all'uso indipendente e completa.
Abb. annuo
6 numeri
lire 15.000
anziché lire ~~18.000~~

ELETTRONICA HOBBY

La rivista per l'hobbista elettronico, il radioamatore, il riparatore radio-TV, l'hardware dei personal computer.
Abb. annuo
12 numeri
lire 32.000
anziché lire ~~42.000~~

NAUTICAL QUARTERLY

Il trimestrale di cultura nautica più prezioso e raffinato del mondo. Emozioni da leggere, guardare e conservare.
Abb. annuo
4 numeri
lire 70.000
anziché lire ~~80.000~~

AUTOMOBILE QUARTERLY

Il trimestrale più prestigioso e raffinato del mondo, dedicato all'auto, alla sua storia, ai suoi miti.
Per chi intende l'auto come un fatto di cultura e di passione.
Abb. annuo
4 numeri
lire 69.500
anziché lire ~~80.000~~

STRUMENTI MUSICALI

Il mensile per i professionisti della musica: audiotest, rassegne, computer music, servizi, interviste.
Abb. annuo
11 numeri
lire 35.000
anziché lire ~~44.000~~

JET SET WILLY II

Sulla scia del successo avuto dal gioco nella sua prima apparizione ecco a voi, come in ogni serie televisiva di successo, la seconda puntata: Jetset willy II.
Il simpatico willy alle prese con nuove sconvolgenti avventure.

IL PROGRAMMA

Per tutti coloro che ci hanno seguito, e quindi già conoscono la precedente versione di Jetset Willy prodotta anch'essa dalla feconda software house inglese, Software Projets, diremo che questo nuovo lavoro è la necessaria evoluzione di un progetto nato parecchio tempo addietro. Jetset Willy infatti, era nato in origine per il computer Spectrum dal quale, malgrado la grandissima abilità di chi ha concepito il programma, ereditava alcune ovvie limitazioni dovute alla memoria utilizzabile e alle capacità sonore.
Come molte volte accade, la limitatezza e le difficoltà incontrate, stimolarono gli sforzi dei progettisti nel semplificare e ridurre le routines di gestione e i dati necessari al programma ai minimi termini, rendendo disponibile già alcuni anni fa un gioco dallo scenario immenso e quindi interminabile.
Interminabile per le infinite situazioni da affrontare ed il divertimento, non certo per il gioco in sé che ha un inizio e (per pochi eletti) anche una vittoriosa risoluzione e fine.
Grazie alle sofisticate tecniche adottate è stato possibile introdurre nel dinamico mondo dei videogames un tipo unico e nuovo di avventura, in cui l'azione si svolge sempre su differenti quadri popolati da esseri in preciso e sincronizzato movimento.
Se la prima versione di Jetset Willy per MSX era un adattamento del famoso game per Spectrum, la nuova versione che vi presentiamo ora, utilizza finalmente appieno le capacità degli MSX beneficiando ugualmente delle tecniche e degli sforzi compiuti per compattare il più possibile il programma.
Utilizzando interamente i 64K di memoria disponibili su MSX per il programma, lo scenario di gioco, nel particolare la casa di Willy, è ora davvero infinita e le stanze si contano ormai su tre cifre.

IL GIOCO

Per molti aspetti il gioco è simile alla precedente versione in quanto a difficoltà e compiti.
Willy ha una casa più grande quindi più oggetti da collezionare, più automi impazziti ostili.
Accanto ad alcune stanze facili da riconoscere come presenti nella prima versione, ne troverete delle nuove con nuove situazioni, oggetti differenti ed ostacoli pericolosi.
Troverete ancora la severa Maria dal piedino nervoso che vi negherà il letto fintantoché non avrete ripulito per bene la casa da tuttociò che visibilmente lampeggia e che dovrete raccogliere. Non ci stancheremo mai di ripetere che ogni schermo ha una propria soluzione, ossia una tattica vincente che consente di raccogliere gli oggetti presenti senza ovviamente lasciarci anche la pelle.

LA TECNICA

Il sistema migliore per un primo approccio con il gioco è imparare a muoversi bene nelle varie stanze, tracciando quando possibile la mappa e prendendo qualche appunto, poi in seguito pensare alla raccolta degli oggetti.
Studiando il difficile compito stanza per stanza, ponendosi pochi problemi per volta, non sarà difficile arrivare dopo qualche tempo alla soluzione finale. Certo il compito è tutt'altro che facile, dobbiamo ammetterlo, però il divertimento anche in questo caso vi farà superare piacevolmente i più alti scogli.
Per riposarvi o riflettere nei momenti più difficili non dimenticate la funzione del tasto "w" che vi consente di fermare momentaneamente il gioco; non potrete certo accomodarvi nel soffice letto preparato dalla signora Maria, no, però potrete sempre utilizzare il vostro...

È JACKSON

LA FORTUNA.

cento e uno premi

1 premio

In palio, fra tutti gli abbonati, una scattante, elegante, ruggente Alfa 33 4x4 giardinetta (1500 cc.) nella versione più prestigiosa e super accessoriata,

perfettamente a suo agio in ogni situazione, potente e nello stesso tempo parca nei consumi, protagonista instancabile di mille avventure.

100 premi

Time of Greenwich®
Dal 1884 segna il tempo dell'Universo.

Il fantastico orologio Time of Greenwich modello 120321. Uno strumento di altissima perfezione ed esclusività, in edizione limitata e personalizzata, realizzata appositamente per i fortunati vincitori abbonati Jackson, dalla prestigiosa Time of Greenwich. Distribuito in Italia da LED ITALY

Regolamento del concorso

1 - Il Gruppo Editoriale Jackson S.p.A. promuove un concorso a premi in occasione della Campagna Abbonamenti 86/87.

2 - Per partecipare è sufficiente sottoscrivere, entro il 31.3.1987, un abbonamento a una delle 20 riviste Jackson.

3 - Sono previsti 1+100 premi da sorteggiare fra tutti gli abbonati.

4 - 1° premio
Un'automobile Alfa 33 4x4 giardinetta (1500 cc.). Gli altri 100 premi consistono in altrettanti orologi Time of Greenwich.

5 - Gli abbonati a più di una rivista avranno diritto, per l'estrazione, all'inserimento del proprio nominativo tante volte quante sono le testate sottoscritte.

6 - L'estrazione dei 1+100 premi in palio avverrà presso la Sede della Jackson entro il 30.5.1987.

7 - L'elenco dei vincitori, ad estrazione avvenuta, sarà pubblicato su almeno 10 delle riviste Jackson.
La vincita inoltre, sarà comunicata con lettera raccomandata a ciascuno dei sorteggiati.

8 - I premi verranno messi a disposizione degli aventi diritto entro 30 giorni dalla data dell'estrazione.

9 - Le spese di immatricolazione della Alfa 33 4x4 giardinetta saranno a carico del vincitore.

10 - I dipendenti, i familiari, i collaboratori del Gruppo Editoriale Jackson sono esclusi dal concorso.

LA TUA RIVISTA.

AUT. MIN. RICH. D.M. n° 4/298449 del 10-12-86

IL PUNTEGGIO

Dato il particolare tipo di gioco, non esiste un conteggio particolare destinato ai "punti", lo scopo è preciso: coricarsi a letto dopo aver raccolto tutti gli oggetti per le varie stanze.
In attesa di riuscire a completare la missione, potete valutare i progressi compiuti, in base al tempo (di sopravvivenza), oggetti raccolti e stanze visitate. Per ciò che riguarda il tempo non vi sono problemi, avete a disposizione solo 9999 ore...

I TASTI DI CONTROLLO

Willy per tradizione può compiere solo tre principali movimenti, il salto e lo spostamento a destra o a sinistra e ciò rende semplicissimo il controllo con tre soli tasti.
Per default è possibile usare i tasti cursore, il joystick, oppure i tasti

della tastiera con la possibilità di ridefinizione. Ciò è stato previsto perché su molte versioni MSX i tasti cursore risultano troppo grossi o scomodi per giocare, e quindi inadatti ad un uso preciso e veloce. Premendo qualsiasi tasto eccetto la barra-spazio, dallo schermo di presentazione, un semplice menù vi permetterà di utilizzare i tasti preferiti per le varie funzioni.
Oltre ai tasti di movimento esistono quattro tasti di controllo:

Q: per diminuire il volume del sottofondo musicale

L: per aumentare il volume del sottofondo musicale

W: per fermare momentaneamente il gioco

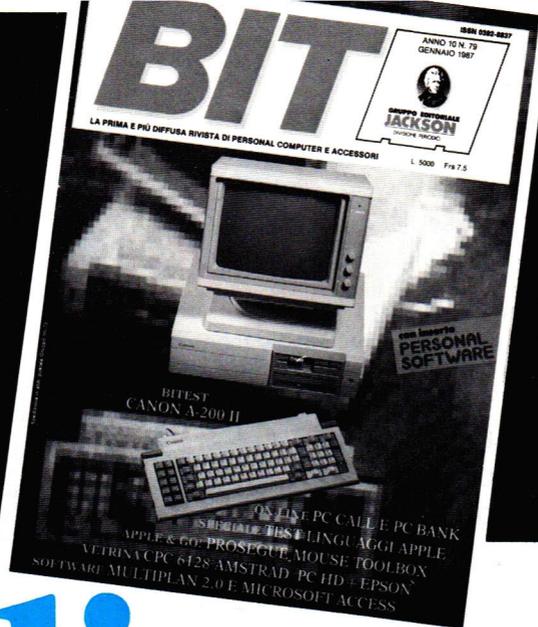
STOP: per uscire dal gioco

IL CARICAMENTO

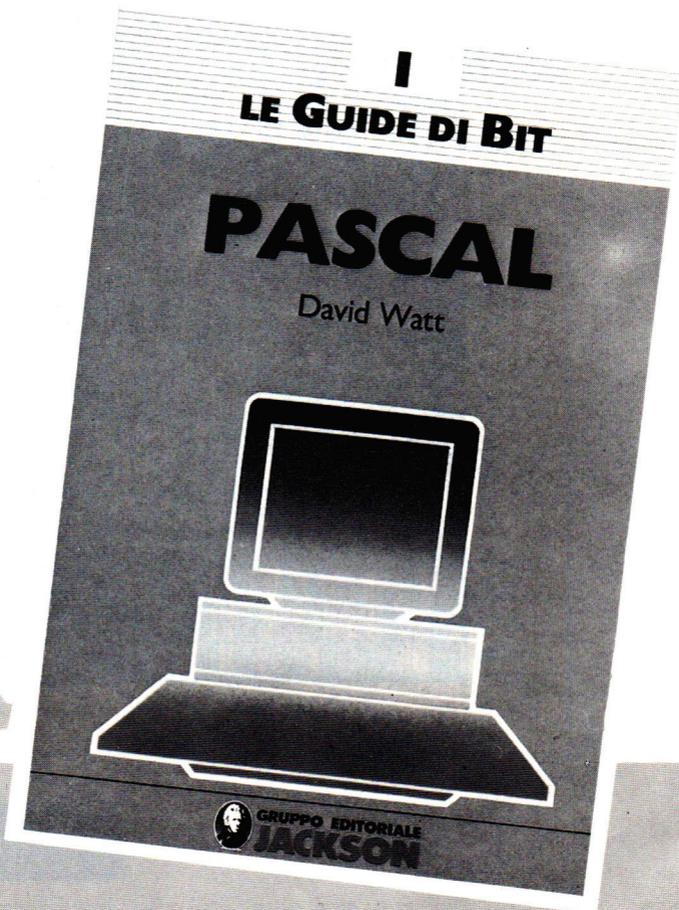
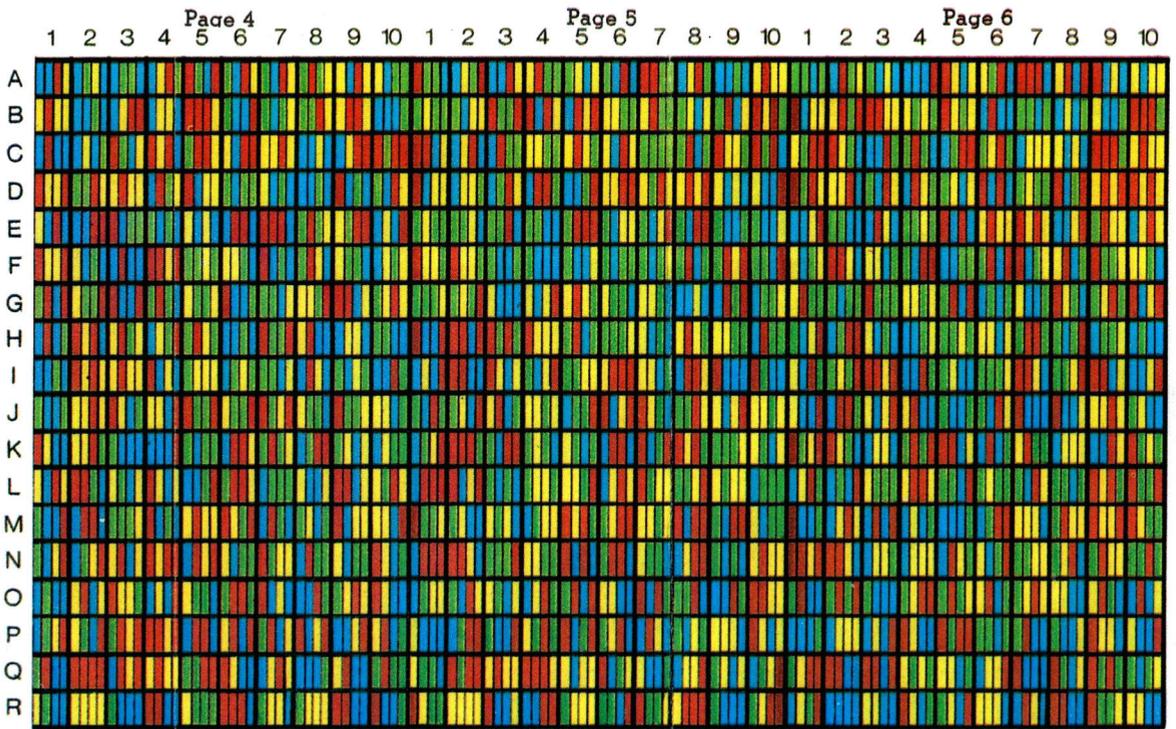
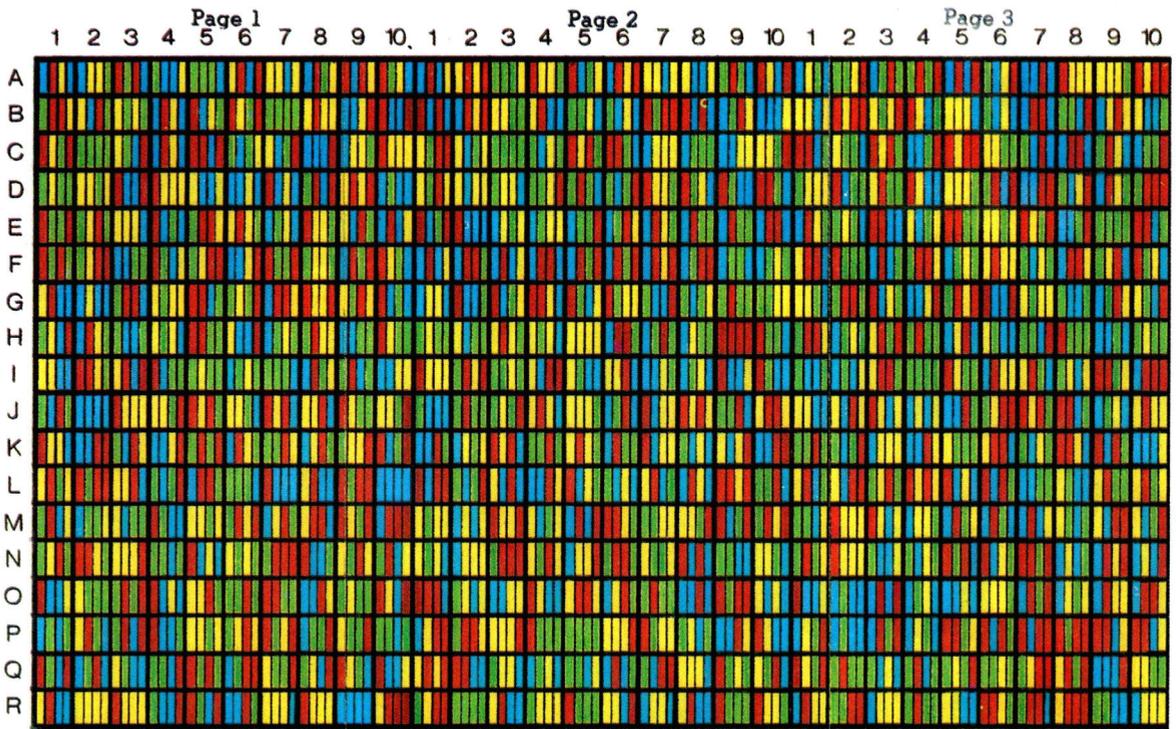
Il gioco Jetset Willy II utilizza appieno le capacità di memoria MSX pertanto può 'girare' solo su quei computer che dispongono di almeno 64K di memoria utente.
Per il caricamento di Jetset Willy II riavvolgere completamente il nastro e battere sul computer:

RUN"cas:"

Un programma iniziale riconoscerà se il vostro computer dispone di 64K di memoria e in caso contrario eviterà le fasi successive dandovi un messaggio appropriato.
Qualora il computer disponga della memoria necessaria, comparirà una schermata di presentazione e la scritta LOADING (caricamento). Una particolare routine di LOAD mostrerà su schermo l'evolversi del caricamento stesso. Al termine il programma partirà autonomamente; qualora invece dovesse apparire un messaggio di "LOAD ERROR" ripetere il caricamento dall'inizio avendo cura di tenere il volume del registratore a 3/4 della corsa complessiva e gli eventuali regolatori di tono a metà corsa.

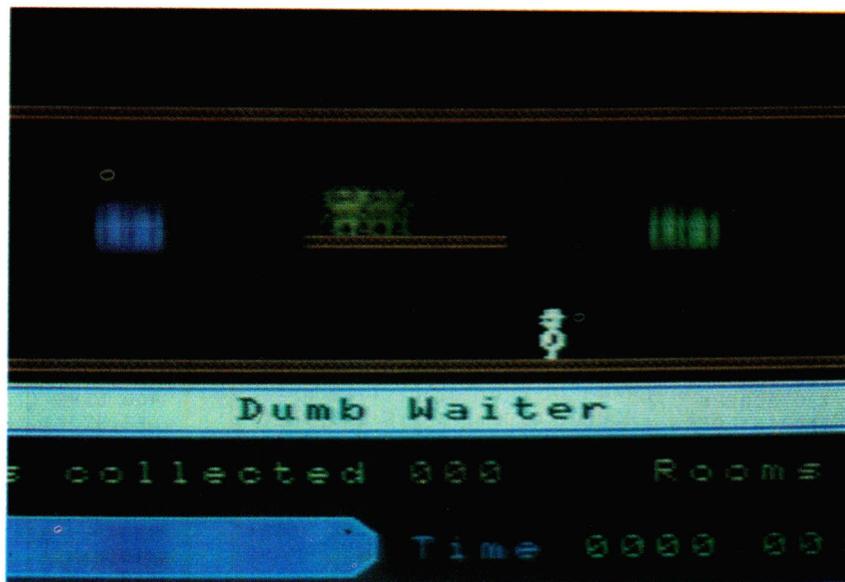
BIT regala

Le Grandi Guide di BIT

Da gennaio e fino a dicembre '87 ogni mese Bit vi da' un grande appuntamento in edicola con le Grandi Guide di Bit: veri manuali di riferimento di eccezionale utilità, dedicati al personal computer, alla programmazione, ai sistemi operativi, ai linguaggi.
A fine anno possederete una biblioteca ricca, aggiornata e completa: per lavorare meglio e conoscere più a fondo il vostro personal computer.

Ogni mese Bit ti aspetta in edicola con la sua Guida.

LA TABELLA DEI COLORI

Anche in Jetset Willy II è stata utilizzata una protezione speciale anti-pirata, consistente in uno speciale codice di accesso al gioco, ricavabile da una particolare tabella a colori.
Solo chi è in possesso della tabella pubblicata in queste pagine potrà giocare una volta inserita la giusta password; conservate quindi con cura questo fascicolo assieme alla cassetta.
Il programma, al termine del caricamento dal nastro, vi chiederà (dopo la pressione della barra-spazio) di immettere un codice di quattro colori da trovare sulla tabella ad un particolare numero di pagina ed alle coordinate specificate.
Sulla tabella converrà cercare il numero di pagina richiesto (può variare tra 1 e 6) e poi la casella corrispondente alle coordinate date.
In questa casella c'è il codice costituito da quattro colori che vi permetterà di accedere al gioco.
Per inserire il codice-colori sul computer, avete a disposizione quattro caselle, in cui dovrete riprodurre la giusta sequenza di colori premendo i tasti da 1 a 4.
Per aiutarvi, lo schermo visualizza in basso a sinistra la corrispondenza tra i tasti numerici e i colori.
Una volta battuto il giusto codice non vi resta che confermare pre-

mendo RETURN. Se il codice inserito è corretto apparirà la schermata iniziale e la musica, altrimenti il codice vi verrà richiesto una seconda volta.
Qualora sbagliaste per due volte consecutive, il programma automaticamente vi lascerà in modo poco garbato ma efficace...

Il Gruppo Editoriale Jackson, proprietario esclusivo dei diritti per l'Italia di questo gioco, invita i lettori che ne fossero a conoscenza, a segnalare l'esistenza di eventuali altre pubblicazioni contenenti questo stesso gioco, alla redazione della nostra rivista.
Tali segnalazioni saranno convenientemente compensate.

GUIDA ALL'INPUT MSX

Per motivi di stampa e chiarezza, i caratteri grafici o comunque tutti i simboli il cui codice (ASC("car")) è superiore a 127, compaiono nei listati codificati in modo particolare.
Ogni carattere o serie di caratteri uguali e contigui, è sostituito da graffe contenenti:

- Il numero dei caratteri (omesso se= 1)
- La lettera maiuscola "C" oppure "A" a seconda se si tratta del set normale (caratteri ottenibili con CHR$(n) o del set alternativo (caratteri ottenuti con CHR$(1)+CHR$(n)).
- Codice "n" del carattere o dei caratteri da battere.

Qualsiasi computer MSX voi abbiate, potete riferirvi al manuale per trovare la corrispondenza tra il codice-carattere dato e il simbolo a cui corrisponde.

ESEMPIO:
{3C190} = battere 3 volte il carattere PI-GRECO
{A67} = battere il carattere cuore.

NORME PER IL CARICAMENTO DEI PROGRAMMI SU CASSETTA

Le istruzioni di caricamento del supergame vengono riportate nel contesto della relativa descrizione. Oltre al supergame sono registrati su nastro tutti gli altri programmi presentati su questo numero i quali vanno caricati, salvo diversa indicazione, con: CLOAD <RETURN>.

LISTATO

AGGUATO SUL FONDO

Occupazione di memoria: 16120 Byte Configurazione richiesta: 32 KByte
Il programma è registrato su cassetta con il nome: AGGUATO

Vi trovate nel fondo del mare e dovrete affrontare due prove diverse. Nella prima dovete aiutare un piccolo pesce a mangiare dei vermetti evitando un grosso e vorace pescecane. Nel secondo siete un subacqueo e dovete colpire lo stesso nemico di prima. Nel primo schermo ogni vermetto mangiato vale un punto, per ogni serie di vermi consumato il punteggio viene moltiplicato per due, ogni arpione tirato (nel secondo schermo) vi fa perdere 50 punti. Ogni pescecane colpito vi varrà 100 punti.
Alla fine del gioco avrete anche una valutazione sulle vostre capacità di pescatore.
Il programma è previsto per funzionare con la tastiera per usare il joistick scrivete "1" alla linea 160 come indicato in listato. Buon divertimento.

```
10 REM ***************************
20 REM *                         *
30 REM *                         *
40 REM *    AGGUATO SUL FONDO    *
50 REM *                         *
60 REM *                         *
70 REM *  di  Maurizio Galluzzo  *
80 REM *                         *
90 REM ***************************
100 REM
110 REM
120 COLOR 15,4,12
130 SCREEN 2,2
140 KEY OFF
150 WIDTH 32
160 OO=0:REM SCRIVERE 0 PER TASTIERA      2
5      REM SCRIVERE 1 PER JOYSTICK
170 GOSUB 3230
180 OPEN "grp:" AS #1
190 LINE (0,128)-(256,192),1,BF
200 LINE (10,133)-(245,185),15,BF
210 PSET (60,140),15:COLOR 1:PRINT #1,"AGG
UATO SUL FONDO"
220 PSET (60,139),15:COLOR 1:PRINT #1,"AGG
UATO SUL FONDO"
230 PSET (48,165),15:COLOR 4:PRINT #1,"di
Maurizio Galluzzo"
240 COLOR 15
250 PSET (0,20),6:DRAW "m0,20m40,20m40,30m
50,30m50,35m40,40m52,43m50,50m60,53m60,60m
40,60m30,53m20,52m10,60m0,60m0,20":PAINT (
20,22),6
260 PSET(0,70),6:DRAW "m0,70m'0,70m20,72m3
0,80m40,90m40,100m50,110m60,110m62,120m70,
127m0,127m0,70":PAINT (10,71),6
270 PSET(256,20),6:DRAW "m230,20m220,30m21
0,30m200,35m190,45m180,50m173,58m170,62m16
8,70m171,80m180,90m178,98m170,105m178,112m
180,120m188,122m190,127m256,127m256,20":PA
INT(240,21),6
280 PSET (77,127),2:DRAW "m80,110m90,80m95
,70m102,60m122,42m140,35m150,32m125,58m101
,81m89,110m86,127m77,127":PAINT (135,40),2
290 PSET (170,30),3:DRAW "m158,32m154,42m1
50,60m143,70m132,78m125,85m119,100m110,114
m100,122m90,127m100,127m110,120m128,100m13
7,85m150,75m156,66m160,50m162,40m170,30":P
AINT (160,35),3
300 RESTORE 3310
310 FOR I%=0 TO 1407
320 READ A:VPOKE 14336+I%,A:NEXT
330 LINE (0,128)-(256,192),1,BF
340 F=RND(-(TIME/10)MOD100)
350 RESTORE 370
360 FOR I=0 TO 7:READ A:VPOKE 4615+I,A:VPO
KE 12807+I,&H81:NEXT
370 DATA 1,1,1,17,42,74,74,132
380 VPOKE 6721,0:VPOKE 6720,0:BEEP:GOSUB 9
50
390 Y=80:X=100:W=8:Z=56:M=1:VI=4:ON INTERV
AL=300 GOSUB 920:ON SPRITE GOSUB 1160
400 LINE (16,0)-(120,8),1,BF:DRAW "bm24,0"
:PRINT #1,"PUNTI:0":LINE (128,0)-(240,8),1
,BF:DRAW "bm136,0":PRINT #1,"VITE:":PUT SP
RITE 28,(176,-5),11,14:PUT SPRITE 29,(200,
-5),11,14:PUT SPRITE 30,(224,-5),11,14
410 PUT SPRITE 0,(W,Z),15,2
420 PUT SPRITE 8,(X,Y),11,10
430 REM
440 REM *************************
450 REM *                       *
460 REM *   CICLO PRINCIPALE    *
470 REM *                       *
480 REM *************************
490 REM
500 A=STICK(OO)
510 IF A=0 THEN 650
520 ON A GOTO 530,540,550,560,570,580,590,
600
530 Y=Y-8:PUT SPRITE 8,(X,Y),11,8:GOTO 610
540 Y=Y-8:X=X+8:PUT SPRITE 8,(X,Y),11,9:GO
TO 610
550 X=X+8:PUT SPRITE 8,(X,Y),11,10:GOTO 61
0
560 Y=Y+8:X=X+8:PUT SPRITE 8,(X,Y),11,11:G
OTO 610
570 Y=Y+8:PUT SPRITE 8,(X,Y),11,12:GOTO 61
0
580 Y=Y+8:X=X-8:PUT SPRITE 8,(X,Y),11,13:G
OTO 610
590 X=X-8:PUT SPRITE 8,(X,Y),11,14:GOTO 61
0
600 Y=Y-8:X=X-8:PUT SPRITE 8,(X,Y),11,15
610 IF X>240 THEN X=240
620 IF X<16 THEN X=16
630 IF Y<16 THEN Y=16
640 IF Y>176 THEN Y=176
650 IF X>W AND Y>Z THEN W=W+M:Z=Z+M:PUT SP
```

```
RITE 0,(W,Z),15,3:GOTO 730
660 IF X>W AND Y=Z THEN W=W+M:PUT SPRITE 0
,(W,Z),15,2:GOTO 840
670 IF X>W AND Y<Z THEN W=W+M:Z=Z-M:PUT SP
RITE 0,(W,Z),15,1:GOTO 750
680 IF Y<Z AND X=W THEN Z=Z-M:PUT SPRITE 0
,(W,Z),15,0:GOTO 840
690 IF X<W AND Y<Z THEN Z=Z-M:W=W-M:PUT SP
RITE 0,(W,Z),15,7:GOTO 790
700 IF X<W AND Y=Z THEN W=W-M:PUT SPRITE 0
,(W,Z),15,6:GOTO 840
710 IF X<W AND Y>Z THEN W=W-M:Z=Z+M:PUT SP
RITE 0,(W,Z),15,5:GOTO 820
720 IF Y>Z AND X=W THEN Z=Z+M:PUT SPRITE 0
,(W,Z),15,4
730 IF W>X THEN W=X
740 IF Z>Y THEN Z=Y
750 GOTO 840
760 IF W>X THEN W=X
770 IF Z<Y THEN Z=Y
780 GOTO 840
790 IF W<X THEN W=X
800 IF Z<Y THEN Z=Y
810 GOTO 840
820 IF W<X THEN W=X
830 IF Z>Y THEN Z=Y
840 INTERVAL ON
850 T=6176+Y*4+X/8
860 IF VPEEK(T)=65 THEN U=U+1:SC=SC+1:GOSU
B 3020
870 IF Y<120 THEN 890
880 VPOKE T,0
890 IF U>23 THEN SC=SC*2:PLAY "v15l64":FOR
 H=96 TO 1 STEP -5:PLAY "n=h;":NEXT:GOSUB
1020
900 SPRITE ON
910 GOTO 500
920 M=M+1
930 IF M>7 THEN M=7
940 RETURN
950 REM
960 REM ************************
970 REM *                      *
980 REM *        VERSO         *
990 REM *                      *
1000 REM ************************
1010 REM
1020 M=0:U=0
1030 FOR J=6659 TO 6851 STEP 64:FOR I=0 TO
 5
1040 Q=INT(RND(1)*26)
1050 IF VPEEK(J+Q)=65 THEN 1040
1060 VPOKE J+Q,65
1070 NEXT:NEXT:IF SC<>0 THEN GOSUB 3020
1080 RETURN
1090 REM
1100 REM ************************
1110 REM *                      *
1120 REM *  COLLISIONE SPRITE   *
1130 REM *                      *
1140 REM ************************
1150 REM
1160 SPRITE OFF:IF TIME<100 THEN SPRITE ON
:RETURN
1170 W=8:Z=56:PUT SPRITE 0,(W,Z),15,2:PUT
SPRITE W,(248,W),0,2:X=248:Y=W*2:VI=VI-1:M
=0:PLAY "l64v15n90v15n85v13n80v12n75v11n65
v10n60v9n55v8n50v7n45v6n40v5n35v4n30v3n25v
2n20v1n15":TIME=M
1180 IF VI=3 THEN PUT SPRITE 30,(224,-5),1
,14
1190 IF VI=2 THEN PUT SPRITE 29,(200,-5),1
,14
1200 IF VI=1 THEN PUT SPRITE 28,(176,-5),1
,14
1210 IF VI=0 THEN 1300
1220 RETURN
1230 REM
1240 REM ************************
1250 REM *                      *
1260 REM *  2. CICLO PRINCIPALE *
1270 REM *                      *
1280 REM ************************
1290 REM
1300 IF SC=0 THEN SC=1
1310 Y=95:X=248:AA=4:G=1:D=1:R=1:S=1:VI=4:
SPRITE OFF:ON STRIG GOSUB 2480,2480
1320 FOR I=4622 TO 4615 STEP -1:VPOKE I,0:
FOR Q=1 TO 600:NEXT:NEXT:PUT SPRITE 30,(22
4,-5),15,6:PUT SPRITE 29,(200,-5),15,6:PUT
 SPRITE 28,(176,-5),15,6
1330 A=STICK(00):ON A GOSUB 1510,1520,1530
,1540,1550,1560,1570,1580
1340 B=B+18:IF B>80 THEN B=0
1350 IF A=0 THEN ON AA GOSUB 1660,1790,192
0,2050
1360 IF S=1 THEN ON G GOSUB 2180,2200,2220
,2240
1370 IF R=1 THEN ON D GOSUB 2330,2350,2370
,2390
1380 IF X>230 THEN X=230
1390 IF X<16 THEN X=16
1400 IF Y<18 THEN Y=18
1410 IF Y>160 THEN Y=160
1420 ON FA GOSUB 2570,2660,2750,2840
1430 STRIG(00) ON
1440 IF F<W+8 AND F>W-8 AND O<Z+8 AND O>Z-
8 THEN GOSUB 2930
1450 IF VI=3 THEN PUT SPRITE 30,(224,-5),1
,6
1460 IF VI=2 THEN PUT SPRITE 29,(200,-5),1
,6
1470 IF VI=1 THEN PUT SPRITE 28,(176,-5),1
,6
1480 IF VI=0 THEN 3210
1490 IF SC<1 THEN 3110
1500 GOTO 1330
1510 Y=Y-2:GOTO 1660
1520 Y=Y-2:X=X+1:GOTO 1790
1530 X=X+2:GOTO 1790
1540 Y=Y+2:X=X+1:GOTO 1790
1550 Y=Y+2:GOTO 1920
1560 Y=Y+2:X=X-1:GOTO 2050
1570 X=X-2:GOTO 2050
1580 Y=Y-2:X=X-1:GOTO 2050
1590 REM
1600 REM ************************
1610 REM *                      *
1620 REM *        ALTO          *
1630 REM *                      *
1640 REM ************************
1650 REM
1660 AA=1:K=1
1670 IF B<20 THEN PUT SPRITE 8,(X,Y),8,16:
PUT SPRITE 9,(X,Y),5,17:PUT SPRITE 10,(X,Y
-15),15,40:RETURN
1680 IF B<40 THEN PUT SPRITE 8,(X,Y),8,18:
PUT SPRITE 9,(X,Y),5,19:PUT SPRITE 10,(X,Y
-15),15,40:RETURN
1690 IF B<60 THEN PUT SPRITE 8,(X,Y),8,20:
PUT SPRITE 9,(X,Y),5,21:PUT SPRITE 10,(X,Y
-15),15,40:RETURN
1700 IF B<80 THEN PUT SPRITE 8,(X,Y),8,18:
PUT SPRITE 9,(X,Y),5,19:PUT SPRITE 10,(X,Y
-15),15,40:RETURN
1710 RETURN
1720 REM
1730 REM ************************
1740 REM *                      *
1750 REM * DX/DX ALTO/DX BASSO  *
1760 REM *                      *
1770 REM ************************
1780 REM
1790 K=2:AA=2
1800 IF B<20 THEN PUT SPRITE 8,(X,Y),8,22:
PUT SPRITE 9,(X,Y),5,23:PUT SPRITE 10,(X+1
5,Y),15,41:RETURN
1810 IF B<40 THEN PUT SPRITE 8,(X,Y),8,24:
PUT SPRITE 9,(X,Y),5,25:PUT SPRITE 10,(X+1
```

```
5,Y),15,41:RETURN
1820 IF B<60 THEN PUT SPRITE 8,(X,Y),8,26:
PUT SPRITE 9,(X,Y),5,27:PUT SPRITE 10,(X+1
5,Y),15,41:RETURN
1830 IF B<80 THEN PUT SPRITE 8,(X,Y),8,24:
PUT SPRITE 9,(X,Y),5,25:PUT SPRITE 10,(X+1
5,Y),15,41:RETURN
1840 RETURN
1850 REM
1860 REM ************************
1870 REM *                      *
1880 REM *        BASSO         *
1890 REM *                      *
1900 REM ************************
1910 REM
1920 K=3:AA=3
1930 IF B<20 THEN PUT SPRITE 8,(X,Y),8,28:
PUT SPRITE 9,(X,Y),5,29:PUT SPRITE 10,(X,Y
+15),15,43:RETURN
1940 IF B<40 THEN PUT SPRITE 8,(X,Y),8,30:
PUT SPRITE 9,(X,Y),5,31:PUT SPRITE 10,(X,Y
+15),15,43:RETURN
1950 IF B<60 THEN PUT SPRITE 8,(X,Y),8,32:
PUT SPRITE 9,(X,Y),5,33:PUT SPRITE 10,(X,Y
+15),15,43:RETURN
1960 IF B<80 THEN PUT SPRITE 8,(X,Y),8,30:
PUT SPRITE 9,(X,Y),5,31:PUT SPRITE 10,(X,Y
+15),15,43:RETURN
1970 RETURN
1980 REM
1990 REM ************************
2000 REM *                      *
2010 REM *  SX/SX BASSO/SX ALTO *
2020 REM *                      *
2030 REM ************************
2040 REM
2050 K=4:AA=K
2060 IF B<20 THEN PUT SPRITE 8,(X,Y),8,34:
PUT SPRITE 9,(X,Y),5,35:PUT SPRITE 10,(X-1
5,Y),15,42:RETURN
2070 IF B<40 THEN PUT SPRITE 8,(X,Y),8,36:
PUT SPRITE 9,(X,Y),5,37:PUT SPRITE 10,(X-1
5,Y),15,42:RETURN
2080 IF B<60 THEN PUT SPRITE 8,(X,Y),8,38:
PUT SPRITE 9,(X,Y),5,39:PUT SPRITE 10,(X-1
5,Y),15,42:RETURN
2090 IF B<80 THEN PUT SPRITE 8,(X,Y),8,36:
PUT SPRITE 9,(X,Y),5,37:PUT SPRITE 10,(X-1
5,Y),15,42:RETURN
2100 RETURN
2110 REM
2120 REM ************************
2130 REM *                      *
2140 REM *   VERTICALE PESCE    *
2150 REM *                      *
2160 REM ************************
2170 REM
2180 IF Y<Z+30 THEN R=0:G=2
2190 RETURN
2200 Z=Z+8:PUT SPRITE 0,(W,Z),15,4:IF Z>(R
ND(1)*50)+120 THEN R=1:G=3
2210 RETURN
2220 IF Y>Z-30 THEN G=4:R=0
2230 RETURN
2240 Z=Z-8:PUT SPRITE 0,(W,Z),15,0:IF Z<(R
ND(1)*16)+54 THEN R=1:G=1
2250 RETURN
2260 REM
2270 REM ************************
2280 REM *                      *
2290 REM *  ORIZZONTALE PESCE   *
2300 REM *                      *
2310 REM ************************
2320 REM
2330 IF X<W+30 THEN D=2:S=0
2340 RETURN
2350 W=W+8:PUT SPRITE 0,(W,Z),15,2:IF W>(R
ND(1)*60)+170 THEN S=1:D=3
2360 RETURN
2370 IF X>W-30 THEN D=4:S=0
2380 RETURN
2390 W=W-8:PUT SPRITE 0,(W,Z),15,6:IF W<(R
ND(1)*64)+16 THEN D=1:S=1
2400 RETURN
2410 REM
2420 REM ************************
2430 REM *                      *
2440 REM *         TIRO         *
2450 REM *                      *
2460 REM ************************
2470 REM
2480 FC=FC+1:F=X:O=Y:PLAY "s10m40000o1a":S
C=SC-50:GOSUB 3020:ON K GOSUB 2570,2660,27
50,2840
2490 RETURN
2500 REM
2510 REM ************************
2520 REM *                      *
2530 REM *      TIRO ALTO       *
2540 REM *                      *
2550 REM ************************
2560 REM
2570 FA=1:PUT SPRITE 11,(F,O-22),15,40:O=O
-10:IF O<16 THEN FA=0:PUT SPRITE 11,(0,0),
0,40:RETURN
2580 RETURN
2590 REM
2600 REM ************************
2610 REM *                      *
2620 REM *      TIRO DESTRA     *
2630 REM *                      *
2640 REM ************************
2650 REM
2660 FA=2:PUT SPRITE 11,(F+22,O),15,41:F=F
+10:IF F>230 THEN FA=0:PUT SPRITE 11,(0,0)
,0,41:RETURN
2670 RETURN
2680 REM
2690 REM ************************
2700 REM *                      *
2710 REM *      TIRO BASSO      *
2720 REM *                      *
2730 REM ************************
2740 REM
2750 FA=3:PUT SPRITE 11,(F,O+22),15,43:O=O
+10:IF O>186 THEN FA=0:PUT SPRITE 11,(0,0)
,0,43:RETURN
2760 RETURN
2770 REM
2780 REM ************************
2790 REM *                      *
2800 REM *     TIRO SINISTRA    *
2810 REM *                      *
2820 REM ************************
2830 REM
2840 FA=4:PUT SPRITE 11,(F-22,O),15,42:F=F
-10:IF F<15 THEN FA=0:PUT SPRITE 11,(0,0),
0,42:RETURN
2850 RETURN
2860 REM
2870 REM ************************
2880 REM *                      *
2890 REM *   MORTE DEL PESCE    *
2900 REM *                      *
2910 REM ************************
2920 REM
2930 VI=VI-1:P=96:FOR N=40 TO 43:PUT SPRIT
E 11,(0,0),0,N:NEXT:FOR I=Z TO 150 STEP 10
:FOR Q=0 TO 7:P=P-1:PLAY "v14n=p;":PUT SPR
ITE 0,(W,I),15,Q:FOR L=0 TO 25:NEXT:NEXT:N
EXT:PUT SPRITE 0,(W,I),15,7:PUT SPRITE 11,
(W-1,I),15,42:FOR I=1 TO 1000:NEXT
2940 SC=SC+100:GOSUB 3020:PS=PS+1:F=0:O=20
9:RETURN
2950 REM
2960 REM ************************
2970 REM *                      *
2980 REM *    STAMPA PUNTI      *
```

```
2990 REM *                        *
3000 REM *************************
3010 REM
3020 IF SC<1 THEN SC=0
3030 LINE (72,0)¬(120,8),1,BF:DRAW "bm64,0
":PRINT #1,SC:PLAY "v15l64o7a":RETURN
3040 REM
3050 REM *************************
3060 REM *                        *
3070 REM *      PERDITA           *
3080 REM *                        *
3090 REM *************************
3100 REM
3110 GOSUB 3230:STRIG(00) OFF:SCREEN 1:IF
SC<1 THEN SC=0:LOCATE 9,2,0:PRINT "PUNTI:"
;SC:LOCATE 0,5,0:PRINT "   NON SEI STATO M
OLTO BRAVO":PRINT:PRINT"   HAI USATO";:PRI
NT USING"###";FC;:PRINT " ARPIONI":PRINT
3120 PRINT "   HAI PRESO ";PS;" PESCI"
3130 FOR I=1 TO 10000:NEXT:RUN
3140 REM
3150 REM *************************
3160 REM *                        *
3170 REM *       VITTORIA         *
3180 REM *                        *
3190 REM *************************
3200 REM
3210 GOSUB 3230:STRIG(00) OFF:SCREEN 1:LOC
ATE 9,2,0:PRINT "PUNTI:";SC:LOCATE 0,6,0:P
RINT " BRAVO ! SEI ARRIVATO ALLA FINE
                                   HAI UCC
ISO 4 PESCI"
3220 LOCATE 0,13,0:PRINT "          CON ";F
C;" ARPIONI":PRINT "
          HAI FATTO UNA BELLA PESCA.":
FOR I=1 TO 10000:NEXT:RUN
3230 SOUND 7,7:SOUND 6,11:SOUND 12,40:FOR
C=8 TO 10:SOUND C,16:NEXT:SOUND 13,14:RETU
RN
3240 REM
3250 REM *************************
3260 REM *                        *
3270 REM *         DATI           *
3280 REM *                        *
3290 REM *************************
3300 REM
3310 DATA 1,2,7,7,7,15,7,7,7,7,3,3,3,1,3,4
,0,128,128,192,224,192,192,192,128,128,192
,128,128,0,128,64
3320 DATA 0,0,0,0,0,0,0,1,1,3,7,61,8,8,0,0
,0,6,10,30,252,124,252,252,244,244,0,0,0,0
,0,0
3330 DATA 0,0,0,0,0,131,95,127,95,132,0,0,
0,0,0,0,0,0,0,0,32,252,254,253,254,248,16,
0,0,0,0,0
3340 DATA 0,0,8,8,56,12,6,15,3,3,3,1,0,1,0
,0,0,0,0,0,0,0,0,128,208,240,240,248,252,2
50,14,0
3350 DATA 2,1,0,1,1,3,1,1,3,3,3,7,3,1,1,0,
32,192,128,192,192,192,224,224,224,224,240
,224,224,224,64,128
3360 DATA 0,0,0,0,0,0,0,1,11,15,15,31,63,9
5,112,0,0,0,16,16,28,48,96,240,192,192,192
,128,0,128,0,0
3370 DATA 0,0,0,0,4,63,127,191,127,31,8,0,
0,0,0,0,0,0,0,0,0,193,250,254,250,33,0,0,0
,0,0,0
3380 DATA 0,96,80,120,63,62,63,63,47,7,0,0
,0,0,0,0,0,0,0,0,0,0,128,128,192,224,188
,16,16,0,0
3390 DATA 0,0,0,0,3,2,7,3,3,3,5,1,3,0,0,0,
0,0,0,0,0,128,128,128,192,128,128,0,128,0,
0,0
3400 DATA 0,0,0,0,0,1,7,3,7,15,61,8,8,0,0,
0,0,0,0,0,0,192,64,192,128,0,0,0,0,0,0
3410 DATA 0,0,0,0,0,0,1,23,31,19,4,0,0,0,0
,0,0,0,0,0,0,0,0,224,208,240,64,0,0,0,0,0
3420 DATA 0,0,4,4,30,7,3,7,1,0,0,0,0,0,0,0
,0,0,0,0,0,64,192,224,160,224,0,0,0,0,0,0
3430 DATA 0,0,0,1,0,1,1,3,1,1,1,0,0,0,0,0,
0,0,0,192,128,160,192,192,192,224,64,192,0
,0,0,0
3440 DATA 0,0,0,0,0,2,3,7,5,7,0,0,0,0,0,0,
0,16,16,60,112,224,192,224,128,0,0,0,0,0,0
,0
3450 DATA 0,0,0,0,0,0,0,7,11,15,2,0,0,0,0,
0,0,0,0,0,0,0,128,232,248,200,32,0,0,0,0,0
3460 DATA 0,0,0,0,0,3,2,3,1,0,0,0,0,0,0,0,
0,0,0,0,0,128,224,192,224,240,188,16,16,0,
0,0
3470 DATA 2,5,5,4,3,1,1,0,0,3,3,6,6,12,8,0
,0,128,128,128,128,128,128,0,0,192,192,96,
96,48,16,0
3480 DATA 0,0,0,0,0,0,0,1,1,0,0,0,0,0,16,1
6,0,64,64,0,0,0,0,128,128,0,0,0,0,0,8,8
3490 DATA 2,5,5,4,3,1,1,0,0,1,1,1,2,2,0,0,
0,128,128,128,128,128,128,0,0,128,64,64,64
,64,0,0
3500 DATA 0,0,0,0,0,0,0,1,1,0,0,0,0,0,4,4,
0,64,64,0,0,0,0,128,128,0,0,0,0,0,32,32
3510 DATA 2,5,5,4,3,1,1,0,0,1,1,1,1,1,0,0,
0,128,128,128,128,128,128,0,0,128,128,128,
128,128,0,0
3520 DATA 0,0,0,0,0,0,0,1,1,0,0,0,0,0,1,1,
0,64,64,0,0,0,0,128,128,0,0,0,0,0,128,128
3530 DATA 0,0,0,0,96,56,30,6,6,30,56,96,0,
0,0,0,0,0,0,0,0,0,126,118,17,14,0,0,0,0,
0
3540 DATA 0,0,0,192,0,0,0,1,1,0,0,0,192,0,
0,0,0,0,0,0,0,0,6,128,128,0,0,0,0,0,0,0
3550 DATA 0,0,0,0,0,0,60,2,14,48,0,0,0,0,0
,0,0,0,0,0,0,0,0,126,118,17,14,0,0,0,0,0
3560 DATA 0,0,0,0,0,192,0,1,1,0,192,0,0,0,
0,0,0,0,0,0,0,0,6,128,128,0,0,0,0,0,0,0
3570 DATA 0,0,0,0,0,0,0,62,62,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,126,118,17,14,0,0,0,0,0
3580 DATA 0,0,0,0,0,0,0,193,193,0,0,0,0,0,
0,0,0,0,0,0,0,0,6,128,128,0,0,0,0,0,0,0
3590 DATA 0,8,12,6,6,3,3,0,0,1,1,3,4,5,5,2
,0,16,48,96,96,192,192,0,0,128,128,128,128
,128,128,0
3600 DATA 16,16,0,0,0,0,0,1,1,0,0,0,0,0,0,
0,8,8,0,0,0,0,0,128,128,0,0,0,0,64,64,0
3610 DATA 0,0,2,2,1,1,1,0,0,1,1,3,4,5,5,2,
0,0,64,64,64,64,128,0,0,128,128,128,128,12
8,128,0
3620 DATA 4,4,0,0,0,0,0,1,1,0,0,0,0,0,0,0,
32,32,0,0,0,0,0,128,128,0,0,0,0,64,64,0
3630 DATA 0,0,1,1,1,1,1,0,0,1,1,3,4,5,5,2,
0,0,128,128,128,128,128,0,0,128,128,128,12
8,128,128,0
3640 DATA 1,1,0,0,0,0,0,1,1,0,0,0,0,0,0,0,
128,128,0,0,0,0,0,128,128,0,0,0,0,64,64,0
3650 DATA 0,0,0,0,0,0,0,126,110,136,112,0,
0,0,0,0,0,0,0,0,6,28,120,96,96,120,28,6,0,
0,0,0
3660 DATA 0,0,0,0,0,0,96,1,1,0,0,0,0,0,0,0
,0,0,0,3,0,0,0,128,128,0,0,0,3,0,0,0
3670 DATA 0,0,0,0,0,0,0,126,110,136,112,0,
0,0,0,0,0,0,0,0,0,0,60,64,112,12,0,0,0,0,0
,0
3680 DATA 0,0,0,0,0,0,96,1,1,0,0,0,0,0,0,0
,0,0,0,0,0,3,0,128,128,0,3,0,0,0,0,0
3690 DATA 0,0,0,0,0,0,0,126,110,136,112,0,
0,0,0,0,0,0,0,0,0,0,124,124,0,0,0,0,0,0,
0
3700 DATA 0,0,0,0,0,0,96,1,1,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,131,131,0,0,0,0,0,0,0
3710 DATA 0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
3720 DATA 0,0,0,0,0,0,0,0,0,252,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
3730 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,63,0,0,0,0,0,0,0
3740 DATA 1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
```

DISASSEMBLER

Configurazione richiesta: 16 KByte + stamp. opz. Occupazione di memoria: 14980 Byte
Il programma è registrato su cassetta con il nome: DISAS

Un disassembler è un programma in grado di trasformare una serie di codici memorizzati nel computer nel corrispettivo significato mnemonico attribuito. Quando si inizia a lavorare con il linguaggio macchina questo strumento diventa indispensabile e serve anche per la traduzione, in un linguaggio comprensibile, delle routine del sistema operativo del computer contenute nelle **ROM.**
I codici prodotti da questo programma sono alcune centinaia. In ogni caso qualsiasi testo che affronti l'utilizzo del microprocessore **Z80** può essere vantaggiosamente utilizzato. Veniamo ora alla spiegazione del programma vero e proprio. Dopo aver caricato il programma con **CLOAD** e averlo mandato in esecuzione con **RUN** compare un menù con tre opzioni. Vediamole in dettaglio.
1. LETTURA MEMORIA Legge una parte della memoria (indifferentemente **RAM** o **ROM**) e riporta i codici relativi.
Bisogna indicare il valore di inizio della area e quello di fine. Naturalmente il valore di fine dovrà essere maggiore di quello di inizio.
Viene chiesto anche se si intende stampare i dati. L'output che si ottiene è il seguente:

INDIRIZZO DEC	HEX	CODICE MACCHINA	CODICE MNEMONICO
0	0	F3	D1
1	1	C3D702	JP 02D7
4	4	BF	CP A
5	5	1B	DEC DE
6	6	98	SBC A,B
7	7	98	SBC A,B
8	8	C38326	JP 2683
11	B	00	NOP
12	C	C3B601	JP 01B6

2. MEMORIZZAZIONE DATI Serve a caricare su RAM un programma in linguaggio macchina che dovrà essere successivamente tradotto. Bisogna indicare l'indirizzo di inizio e poi in sequenza i codici (da 0 a 255) per terminare scrivere **FINE.**
3. FINE PROGRAMMA Termina il programma ma solo dopo aver confermato l'opzione.

```
10 REM  *****************************
20 REM  *                           *
30 REM  *     DISASSEMBLER  Z80     *
40 REM  *                           *
50 REM  *                           *
60 REM  *   di Maurizio Galluzzo    *
70 REM  *                           *
80 REM  * GRUPPO EDITORIALE JACKSON *
90 REM  *                           *
100 REM *                           *
110 REM *****************************
120 REM
130 REM
140 REM
150 CLEAR 400
160 KEY OFF
170 SCREEN 0
180 WIDTH 40
190 COLOR 15,1
200 CLS
210 ST=0
220 POKE &HFCAB,255
230 OUT &HAB,12
240 GOSUB 2340
250 LOCATE 11,3
260 PRINT "DISASSEMBLER Z80"
270 LOCATE 9,5
280 PRINT "di Maurizio Galluzzo"
290 LOCATE 4,7
300 PRINT "Gruppo Editoriale Jackson 1986'
310 PLAY "dcd"
320 LOCATE 4,19
330 PRINT "Premi un tasto per continuare"
340 GOSUB 1070
350 CLS
360 GOTO 2120
370 REM
380 REM ***********************
390 REM *                     *
400 REM *   LETTURA MEMORIA   *
410 REM *                     *
420 REM ***********************
430 REM
440 CLS
450 LOCATE 11,3
460 PRINT "DISASSEMBLER Z80"
470 PRINT:PRINT:PRINT
480 INPUT "      Inizio area : ";A$
490 A=VAL(A$)
500 INPUT "      Fine area   : ";B$
510 W=A
520 B=VAL(B$):W=A
530 IF B<=A THEN BEEP:PRINT:PRINT "      --
--------- Errore ------------":PRINT:PRINT "
     CONTROLLA I DATI":PRINT:GOTO 480
540 PRINT:PRINT:PRINT"     VUOI STAMPARE <
S/N> ?":PRINT:PRINT
550 ST$=INKEY$:IF ST$="" THEN 550
560 IF ST$="S" THEN PRINT "PREPARA LA STAM
PANTE poi premi un tasto":PRINT:PRINT:ST=1
:GOSUB 1070
570 GOSUB 1060
580 PI=1
```

```
590 IF ST=1 THEN GOSUB 1080
600 C1$=HEX$(PEEK(W))
610 IF LEN(C1$)=1 THEN C1$="0"+C1$
620 C2$=HEX$(PEEK(W+1))
630 IF LEN(C2$)=1 THEN C2$="0"+C2$
640 C3$=HEX$(PEEK(W+2))
650 IF LEN(C3$)=1 THEN C3$="0"+C3$
660 C4$=HEX$(PEEK(W+2))
670 IF LEN(C4$)=1 THEN C4$="0"+C4$
680 IF C1$="CB" THEN RESTORE 1390:GOTO 760
690 IF C1$="ED" THEN RESTORE 1570:GOTO 760
700 IF C1$="FD" AND C2$="CB" THEN RESTORE
1630:GOTO 780
710 IF C1$="DD" AND C2$="CB" THEN RESTORE
1630:D=1:GOTO 780
720 IF C1$="FD" THEN RESTORE 1670:GOTO 810
730 IF C1$="DD" THEN RESTORE 1670:D=1:GOTO
 810
740 RESTORE 1180
750 GOSUB 840:IF MID$(X$,1,2)=C1$ THEN 900
 ELSE GOTO 750
760 GOSUB 840:IF MID$(X$,1,2)=C2$ THENX$=C
1$+X$:GOTO 900
770 GOTO 760
780 GOSUB 840
790 IF MID$(X$,1,2)=C4$ THENX$=C1$+C2$+C3$
+X$:GOTO 900
800 GOTO 780
810 IF C2$="36" THEN X$=C1$+C2$+C3$+C4$:Y$
="LD (IY+"+C3$+")."+C4$:Z1=4:GOTO 960
820 GOSUB 840:IF MID$(X$,1,2)=C2$ THEN X$=
C1$+X$:GOTO 900
830 GOTO 810
840 READ X$
850 READ Y$
860 READ Z$
870 IF X$="mmm" AND ST=1 THEN GOSUB 1090:L
PRINT A;TAB(9);HEX$(W);TAB(15);C1$;" --¬ Z
ONA DI DATI ¬¬-"
880 IF X$="mmm" THEN GOSUB 1090:PRINT A;TA
B(9);HEX$(W);TAB(15);C1$;" --- ZONA DI DAT
I ¬¬¬":W=W+1:RETURN 600
890 RETURN
900 Z1=VAL(MID$(Z$,1,1)):Z2=VAL(MID$(Z$,2,
1)):Z3=VAL("&H"+(MID$(Z$,3,1))):IF Z2=0 TH
EN 970
910 IF Z2=2 THEN CD$=C2$:CM$=C2$:CL=2
920 IF Z2=3 THEN CD$=C3$:CM$=C3$:CL=2
930 IF Z2=4 THEN CD$=C3$+C2$:CM$=C2$+C3$:C
L=4:Z2=2
940 IF Z2=5 THEN CD$=C4$+C3$:CM$=C3$+C4$:C
L=4:Z2=3
950 MID$(X$,(Z2*2)¬1,CL)=CM$:MID$(Y$,Z3,CL
)=CD$
960 IF D=1 THEN GOSUB 1040
970 GOSUB 1020:GOSUB 1090:PRINT A;TAB(9);H
EX$(W);TAB(15);X$;TAB(25);Y$:RI=RI+1:IF RI
=20 THEN PLAY"DC":GOSUB 1070:CLS:GOSUB 106
0:RI=1
980 IF ST=1 THEN LPRINT A;TAB(9);HEX$(W);T
AB(15);X$;TAB(25);Y$
990 W=W+Z1:IF W>=B THEN PRINT:GOSUB 1090:P
RINT A;TAB(9);HEX$(W);TAB(15);"FINE DELL'A
REA":PRINT:IF ST=0 THEN GOSUB 1070:GOTO 21
20
1000 IF W>=B AND ST=1 THEN LPRINT:LPRINT A
;TAB(9);HEX$(W);TAB(15);"FINE DELL'AREA":L
PRINT:GOSUB 1070:GOTO 2120
1010 GOTO 600
1020 FOR Q=1 TO LEN(Y$):IF MID$(Y$,Q,1)=CH
R$(46) THEN MID$(Y$,Q,1)=CHR$(44)
1030 NEXT:RETURN
1040 FOR QQ=1 TO LEN(Y$):IF MID$(Y$,QQ,1)=
CHR$(89) THEN MID$(Y$,QQ,1)=CHR$(88)
1050 NEXT:RETURN
1060 CLS:PRINT:PRINT "{C255}¬¬INDIRIZZO¬¬
 CODICE   CODICE ":PRINT "DECIM.  ESAD.
      MNEMONICO":PRINT:RETURN
1070 IF INKEY$="" THEN 1070 ELSE RETURN
1080 LPRINT:LPRINT "¬¬INDIRIZZO¬-  CODICE
 CODICE ":LPRINT "DECIM.  ESAD.
MNEMONICO":LPRINT:RETURN
1090 IF SGN(W)=¬ THEN A=32768!-(ABS(W)-327
68!) ELSE A=W
1100 RETURN
1110 REM
1120 REM ************************
1130 REM *                      *
1140 REM *        DATI          *
1150 REM *                      *
1160 REM ************************
1170 REM
1180 DA{C255}TA 8E,ADC A.(HL),1,8F,ADC A.A
,1,88,ADC A.B,1,89,ADC A.C,1,8A,ADC A.D,1,
8B,ADC A.E,1,8C,ADC A.H,1,8D,ADC A.L,1,CEn
n,ADC A.NN,227,86,ADD A.(HL),1,87,ADD A.A,
1,80,ADD A.B,1
1190 DATA 81,ADD A.C,1,82,ADD A.D,1,83,ADD
 A.E,1,84,ADD A.H,1,85,ADD A.L,1,C6nn,ADD
.NN,227,09,ADD HL.BC,1,19,ADD HL.DE,1,29,A
DD HL.HL,1,39,ADD HL.SP,1,A6,AND (HL),1,A7
,AND A,1
1200 DATA A0,AND B,1,A1,AND C,1,A2,AND D,1
,A3,AND E,1,A4,AND H,1,A5,AND L,1,E6nn,AND
 NN,225,D9,EXX,1,76,HALT,1,DBnn,IN A.(NN),
227,DCaaaa,CALL C.ADRR,348,FCaaaa,CALL M.A
DRR,348
1210 DATA D4aaaa,CALL NC.ADRR,349,CDaaaa,C
ALL ADRR,346,C4aaaa,CALL NZ.ADRR,349,F4aaa
a,CALL P.ADRR,348,ECaaaa,CALL PE.ADRR,349,
E4aaaa,CALL PO.ADRR,349,CCaaaa,CALL Z.ADRR
,348
1220 DATA 3F,CCF,1,BE,CP (HL),1,BF,CP A,1,
B8,CP B,1,B9,CP C,1,BA,CP D,1,BB,CP E,1,BC
,CP H,1,BD,CP L,1,FEnn,CP NN,224,2F,CPL,1,
27,DAA,1
1230 DATA 35,DEC (HL),1,3D,DEC A,1,05,DEC
B,1,0B,DEC BC,1,0D,DEC C,1,15,DEC D,1,1B,D
EC DE,1,1D,DEC E,1,25,DEC H,1,2B,DEC HL,1,
2D,DEC L,1,3B,DEC SP,1,F3,DI,1,10dd,DJNZ d
e,226
1240 DATA FB,EI,1,E3,EX (SP).HL,1,08,EX AF
.AF,1,EB,EX DE.HL,1,34,INC (HL),1,3C,INC A
,1,04,INC B,1,03,INC BC,1,0C,INC C,1,14,IN
C D,1,13,INC DE,1,1C,INC E,1,24,INC H,1
1250 DATA 23,INC HL,1,2C,INC L,1,33,INC SP
,1,E9,JP (HL),1,DAaaaa,JP C.ADRR,346,FAaaa
a,JP M.ADRR,346,D2aaaa,JP NC.ADRR,347,C3aa
aa,JP ADRR,344,C2aaaa,JP NZ.ADRR,347,F2aaa
a,JP P.ADRR,346
1260 DATA EAaaaa,JP PE.ADRR,347,E2aaaa,JP
PO.ADRR,347,CAaaaa,JP Z.ADRR,346,38dd,JR C
.de,226,18dd,JR de,224,30dd,JR NC.de,227,2
0dd,JR NZ.de,227,28dd,JR Z.de,226,02,LD (B
C).A,1
1270 DATA 12,LD (DE).A,1,77,LD (HL).A,1,70
,LD (HL).B,1,71,LD (HL).C,1,72,LD (HL).D,1
,73,LD (HL).E,1,74,LD (HL).H,1,75,LD (HL).
L,1,36nn,LD (HL).NN,229,4C,LD C.H,1,4D,LD
C.L,1
1280 DATA 0Enn,LD C.NN,226,56,LD D.(HL),1,
57,LD D.A,1,50,LD D.B,1,51,LD D.C,1,52,LD
D.D,1,53,LD D.E,1,54,LD D.H,1,55,LD D.L,1,
32nnnn,LD (NNNN).A,345,22nnnn,LD (NNNN).HL
,345
1290 DATA 0A,LD A.(BC),1,1A,LD A.(DE),1,7E
,LD A.(HL),1,3Annnn,LD A.(NNNN),347,7F,LD
A.A,1,78,LD A.B,1,79,LD A.C,1,7A,LD A.D,1,
7B,LD A.E,1,7C,LD A.H,1,7D,LD A.L,1,3Enn,L
D A.NN,226,46,LD B.(HL),1,47,LD B.A,1,40,L
D B.B,1,41,LD B.C,1,42,LD B.D,1,43,LD B.E,
1
1300 DATA 44,LD B.H,1,45,LD B.L,1,06nn,LD
B.NN,226,01nnnn,LD BC.NNNN,347,4E,LD C.(HL
),1,4F,LD C.A,1,48,LD C.B,1,49,LD C.C,1,4A
,LD C.D,1,4B,LD C.E,1,31nnnn,LD SP.NNNN,34
7,00,NOP,1
1310 DATA B6,OR (HL),1,B7,OR A,1,B0,OR B,1
```

```
,B1,OR C,1,B2,OR D,1,B3,OR E,1,B4,OR H,1,B
5,OR L,1,F6nn,OR NN,224,16nn,LD D.NN,226,1
1nnnn,LD DE.NNNN,347,5E,LD E.(HL),1,5F,LD
E.A,1
1320 DATA 58,LD E.B,1,59,LD E.C,1,5A,LD E.
D,1,5B,LD E.E,1,5C,LD E.H,1,5D,LD E.L,1,1E
nn,LD E.NN,226,66,LD H.(HL),1,67,LD H.A,1,
60,LD H.B,1,61,LD H.C,1,62,LD H.D,1,63,LD
H.E,1,64,LD H.H,1
1330 DATA 65,LD H.L,1,26nn,LD H.NN,226,2An
nnn,LD HL.(NNNN),348,21nnnn,LD HL.NNNN,347
,6E,LD L.(HL),1,6F,LD L.A,1,68,LD L.B,1,69
,LD L.C,1,6A,LD L.D,1,6B,LD L.E,1,6C,LD L.
H,1,6D,LD L.L,1
1340 DATA 2Enn,LD L.NN,226,F9,LD SP.HL,1,D
3nn,OUT (NN).A,226,F1,POP AF,1,C1,POP BC,1
,D1,POP DE,1,E1,POP HL,1,F5,PUSH AF,1,C5,P
USH BC,1,D5,PUSH DE,1,E5,PUSH HL,1,C9,RET,
1
1350 DATA D8,RET C,1,F8,RET M,1,D0,RET NC,
1,C0,RET NZ,1,F0,RET P,1,E8,RET PE,1,E0,RE
T PO,1,C8,RET Z,1,17,RLA,1,C7,RST 00,1,CF,
RST 08,1,D7,RST 10H,1,DF,RST 18H,1,E7,RST
20H,1,EF,RST 28H,1,F7,RST 30H,1,FF,RST 38H
,1,9E,SBC A.(HL),1
1360 DATA 9F,SBC A.A,1,98,SBC A.B,1,99,SBC
 A.C,1,9A,SBC A.D,1,9B,SBC A.E,1,9C,SBC A.
H,1,9D,SBC A.L,1,DEnn,SBC A.NN,227,37,SCF,
1,07,RLCA,1,1F,RRA,1,0F,RRCA,1,91,SUB C,1,
92,SUB D,1,93,SUB E,1,94,SUB H,1
1370 DATA 95,SUB L,1,D6nn,SUB NN,225,AE,XO
R (HL),1,AF,XOR A,1,A8,XOR B,1,A9,XOR C,1,
AA,XOR D,1,AB,XOR E,1,AC,XOR H,1,AD,XOR L,
1,EEnn,XOR NN,225,96,SUB (HL),1,97,SUB A,1
,90,SUB B,1
1380 DATA mmm,,
1390 DATA 46,BIT 0.(HL),2,47,BIT 0.A,2,40,
BIT 0.B,2,41,BIT 0.C,2,42,BIT 0.D,2,43,BIT
 0.E,2,44,BIT 0.H,2,45,BIT 0.L,2,4E,BIT 1.
(HL),2,7F,BIT 7.A,2,78,BIT 7.B,2,79,BIT 7.
C,2,7A,BIT 7.D,2,7B,BIT 7.E,2,4F,BIT 1.A,2
,48,BIT 1.B,2
1400 DATA 49,BIT 1.C,2,4A,BIT 1.D,2,4B,BIT
 1.E,2,4C,BIT 1.H,2,4D,BIT 1.L,2,56,BIT 2.
(HL),2,57,BIT 2.A,2,50,BIT 2.B,2,51,BIT 2.
C,2,52,BIT 2.D,2,53,BIT 2.E,2,54,BIT 2.H,2
,55,BIT 2.L,2,5E,BIT 3.(HL),2,5F,BIT 3.A,2
,58,BIT 3.B,2,59,BIT 3.C,2
1410 DATA 5A,BIT 3.D,2,5B,BIT 3.E,2,5C,BIT
 3.H,2,5D,BIT 3.L,2,66,BIT 4.(HL),2,67,BIT
 4.A,2,60,BIT 4.B,2,61,BIT 4.C,2,62,BIT 4.
D,2,63,BIT 4.E,2,64,BIT 4.H,2,65,BIT 4.L,2
,6E,BIT 5.(HL),2,6F,BIT 5.A,2
1420 DATA 68,BIT 5.B,2,69,BIT 5.C,2,6A,BIT
 5.D,2,6B,BIT 5.E,2,6C,BIT 5.H,2,6D,BIT 5.
L,2,76,BIT 6.(HL),2,77,BIT 6.A,2,70,BIT 6.
B,2,71,BIT 6.C,2,72,BIT 6.D,2,73,BIT 6.E,2
,74,BIT 6.H,2
1430 DATA 75,BIT 6.L,2,7E,BIT 7.(HL),2,7C,
BIT 7.H,2,7D,BIT 7.L,2,8B,RES 1.E,2,8C,RES
 1.H,2,8D,RES 1.L,2,96,RES 2.(HL),2,97,RES
 2.A,2,90,RES 2.B,2,91,RES 2.C,2,92,RES 2.
D,2,93,RES 2.E,2
1440 DATA 94,RES 2.H,2,95,RES 2.L,2,9E,RES
 3.(HL),2,9F,RES 3.A,2,98,RES 3.B,2,99,RES
 3.C,2,86,RES 0.(HL),2,87,RES 0.A,2,80,RES
 0.B,2,81,RES 0.C,2,82,RES 0.D,2,83,RES 0.
E,2,84,RES 0.H,2,85,RES 0.L,2,8E,RES 0.(HL
),2
1450 DATA 8F,RES 1.A,2,88,RES 1.B,2,89,RES
 1.C,2,8A,RES 1.D,2,BB,RES 7.E,2,BC,RES 7.
H,2,BD,RES 7.L,2,16,RL (HL),2,17,RL A,2,10
,RL.B,2,11,RL C,2,12,RL D,2,13,RL E,2,14,R
L H,2,15,RL L,2
1460 DATA 9A,RES 3.D,2,9B,RES 3.E,2,9C,RES
 3.H,2,9D,RES 3.L,2,A6,RES 4.(HL),2,A7,RES
 4.A,2,A0,RES 4.B,2,A1,RES 4.C,2,A2,RES 4.
D,2,A3,RES 4.E,2,A4,RES 4.H,2,A5,RES 4.L,2
,AE,RES 5.(HL),2
1470 DATA AF,RES 5.A,2,A8,RES 5.B,2,A9,RES
 5.C,2,AA,RES 5.D,2,AB,RES 5.E,2,AC,RES 5.
H,2,AD,RES 5.L,2,B6,RES 6.(HL),2,B7,RES 6.
A,2,B0,RES 6.B,2,B1,RES 6.C,2,B2,RES 6.D,2
,B3,RES 6.E,2,B4,RES 6.H,2
1480 DATA B5,RES 6.L,2,BE,RES 7.(HL),2,BF,
RES 7.A,2,B8,RES 7.B,2,B9,RES 7.C,2,BA,RES
 7.D,2,C6,SET 0.HL,2,06,RLC (HL),2,07,RLC
A,2,00,RLC B,2,01,RLC C,2,02,RLC D,2,03,RL
C E,2
1490 DATA 04,RLC H,2,05,RLC L,2,1E,RR (HL)
,2,1F,RR A,2,18,RR B,2,19,RR C,2,1A,RR D,2
,1B,RR E,2,1C,RR H,2,1D,RR L,2,0E,RRC (HL)
,2,0F,RRC A,2,08,RRC B,2,09,RRC C,2,0A,RRC
 D,2,0B,RRC E,2,0C,RRC H,2,0D,RRC L,2
1500 DATA DA,SET 3.D,2,DB,SET 3.E,2,DC,SET
 3.H,2,DD,SET 3.L,2,E6,SET 4.(HL),2,E7,SET
 4.A,2,E0,SET 4.B,2,E1,SET 4.C,2,E2,SET 4.
D,2,E3,SET 4.E,2,E4,SET 4.H,2,E5,SET 4.L,2
,EE,SET 5.(HL),2,EF,SET 5.A,2
1510 DATA E8,SET 5.B,2,E9,SET 5.C,2,EA,SET
 5.D,2,EB,SET 5.E,2,EC,SET 5.H,2,ED,SET 5.
L,2,F6,SET 6.(HL),2,F7,SET 6.A,2,F0,SET 6.
B,2,F1,SET 6.C,2,F2,SET 6.D,2,C7,SET 0.A,2
,C0,SET 0.B,2,C1,SET 0.C,2,C2,SET 0.D,2
1520 DATA C3,SET 0.E,2,C4,SET 0.H,2,C5,SET
 0.L,2,CE,SET 1.(HL),2,CF,SET 1.A,2,C8,SET
 1.B,2,C9,SET 1.C,2,CA,SET 1.D,2,CB,SET 1.
E,2,CC,SET 1.H,2,CD,SET 1.L,2,D6,SET 2.(HL
),2,D7,SET 2.A,2,D0,SET 2.B,2
1530 DATA D1,SET 2.C,2,D2,SET 2.D,2,D3,SET
 2.E,2,D4,SET 2.H,2,D5,SET 2.L,2,DE,SET 3.
(HL),2,DF,SET 3.A,2,D8,SET 3.B,2,D9,SET 3.
C,2,20,SLA B,2,21,SLA C,2,22,SLA D,2,23,SL
A E,2,24,SLA H,2,25,SLA L,2,2E,SRA (HL),2,
2F,SRA A,2
1540 DATA 28,SRA B,2,29,SRA C,2,2A,SRA D,2
,2B,SRA E,2,2C,SRA H,2,2D,SRA L,2,3E,SRL (
HL),2,3F,SRL A,2,38,SRL B,2,39,SRL C,2,3A,
SRL D,2,3B,SRL E,2,3C,SRL H,2,3D,SRL L,2,F
3,SET 6.E,2
1550 DATAF4,SET 6.H,2,F5,SET 6.L,2,FE,SET
7.(HL),2,FF,SET 7.A,2,F8,SET 7.B,2,F9,SET
7.C,2,FA,SET 7.D,2,FB,SET 7.E,2,FC,SET 7.H
,2,FD,SET 7.L,2,26,SLA (HL),2,27,SLA A,2
1560 DATA mmm,,
1570 DATA 4A,ADC HL.BC,2,5A,ADC HL.DE,2,6A
,ADC HL.HL,2,7A,ADC HL.SP,2,46,IM 0,2,56,I
M 1,2,5E,IM 2,2,78,IN A.(C),2,A9,CPD,2,B9,
CPDR,2,A1,CPI,2,B1,CPIR,2,40,IN B.(C),2
1580 DATA 48,IN C.(C),2,50,IN D.(C),2,58,I
N E.(C),2,60,IN H.(C),2,68,IN L.(C),2,AA,I
ND,2,BA,INDR,2,A2,INI,2,B2,IMIR,2,57,LD A.
I,2,5F,LD A.R,2,A8,LDD,2,B8,LDDR,2,A0,LDI,
2,B0,LDIR,2,44,NEG,2,47,LD I.A,2
1590 DATA 4F,LD R.A,2,BB,OTDR,2,B3,OTIR,2,
79,OUT (C).A,2,41,OUT (C).B,2,49,OUT (C).C
,2,51,OUT (C).D,2,59,OUT (C).E,2,61,OUT (C
).H,2,69,OUT (C).L,2,AB,OUTD,2
1600 DATA A3,OUTI,2,4D,RETI,2,45,RETN,2,42
,SBC HL.BC,2,52,SBC HL.DE,2,62,SBC HL.HL,2
,72,SBC HL.SP,2,6F,RLD,2,67,RRD,2,43nnnn,L
D (NNNN).BC,455,53nnnn,LD (NNNN).DE,455,73
nnnn,LD (NNNN),455,4Bnnnn,LD BC.(NNNN),458
1610 DATA 5Bnnnn,LD DE.(NNNN),458,7Bnnnn,L
D SP.(NNNN),458
1620 DATA mmm,,
1630 DATA 46,BIT 0.(IY+dd),43B,7E,BIT 7.(I
Y+dd),43B,4E,BIT 1.(IY+dd),43B,56,BIT 2.(I
Y+dd),43B,5E,BIT 3.(IY+dd),43B,66,BIT 4.(I
Y+dd),43B,6E,BIT 5.(IY+dd),43B,76,BIT 6.(I
Y+dd),43B
1640 DATA 96,RES 2.(IY+dd),43B,9E,RES 3.(I
Y+dd),43B,86,RES 0.(IY+dd),43B,8E,RES 1.(I
Y+dd),43B,16,RL (IY+dd),438,A6,RES 4.(IY+d
d),43B,AE,RES 5.(IY+dd),43B,B6,RES 6.(IY+d
d),43B,BE,RES 7.(IY+dd),43B,06,RLC (IY+dd)
```

```
,439,1E,RR (IY+dd),438,0E,RRC (IY+dd),439
1650 DATA E6,SET 4.(IY+dd),43B,EE,SET 5.(I
Y+dd),43B,F6,SET 6.(IY+dd),43B,C6,SET 0.(I
Y+dd),43B,CE,SET 1.(IY+dd),43B,D6,SET 2.(I
Y+dd),43B,DE,SET 3.(IY+dd),43B,2E,SRA (IY+
dd),439,3E,SRL (IY+dd),439,FE,SET 7.(IY+dd
),43B,26,SLA (IY+dd),439
1660 DATA mmm,,
1670 DATA 8Edd,ADC A.(IY+dd),33B,86dd,ADD
A.(IY+dd),33B,09,ADD IY.BC,2,19,ADD IY.DE,
2,29,ADD IY.IY,2,39,ADD IY.SP,2,A6dd,AND (
IY+dd),339,BEdd,CP (IY+dd),338,35dd,DEC (I
Y+dd),339
1680 DATA 2B,DEC IY,2,E3,EX (SP).IY,2,77dd
,LD (IY+dd).A,338,70dd,LD (IY+dd).B,338,71
dd,LD (IY+dd).C,338,72dd,LD (IY+dd).D,338,
73dd,LD (IY+dd),338,34dd,INC (IY+dd),339,2
3,INC IY,2
1690 DATA E9,JP (IY),2,56dd,LD D.(IY+dd),3
3A,74dd,LD (IY+dd).H,338,75dd,LD (IY+dd).L
,338,22nnnn,LD (NNNN).IY,455,7Edd,LD A.(IY
+dd),33A,46dd,LD B.(IY+dd),33A,4Edd,LD C.(
IY+dd),33A,F9,LD SP.IY,2
1700 DATA B6dd,OR (IY+dd),338,5Edd,LD E.(I
Y+dd),33A,66dd),LD H.(IY+dd),33A,2Annnn,LD
 IY.(NNNN),458,21nnnn,LD IY.NNNN,457,6Edd,
LD L.(IY+dd),33A,E1,POP IY,2,E5,PUSH IY,2,
9Edd,SBC A.(IY+dd),33B,96dd,SUB (IY+dd),33
9,AEdd,XOR (IY+dd),339
1710 DATA mmm,,,kkk,,
1720 RESTORE:W=0
1730 W=W+1
1740 READ X$
1750 READ Y$
1760 READ Z$
1770 IF X$="kkk" THEN END
1780 PRINT X$;TAB(10);Y$;TAB(25);Z$;TAB(30
);W
1790 GOTO 1730
1800 REM
1810 REM ***********************
1820 REM *                     *
1830 REM *  INSERIMENTO DATI   *
1840 REM *                     *
1850 REM ***********************
1860 REM
1870 ON ERROR GOTO 2570
1880 CLS
1890 PRINT
1900 PRINT "                INSERIMENTO DATI"
1910 PRINT:PRINT:PRINT:PRINT
1920 PRINT "        Indirizzo inizio ";:INPU
T A$
1930 A=VAL(A$):LL=A
1940 IF A>65535! OR A<0 THEN BEEP:BEEP:GOT
O 1920
1950 PRINT "        Scrivi <FINE> per termin
are"
1960 PRINT:PRINT
1970 FOR G=A TO 65535!
1980 PRINT "        ";USING "#####";G;:PRINT
 " --> ";:LINE INPUT A$
1990 IF A$="FINE" THEN 2040
2000 A=VAL(A$)
2010 IF A<0 OR A>255 THEN PRINT:PRINT "
              ERRORE":PRINT:BEEP:BEEP:GOTO 1
980
2020 POKE G,A
2030 NEXT
2040 PRINT:PRINT
2050 PRINT "        Indirizzo inizio : ";LL
2060 PRINT "        Indirizzo fine   : ";G-1
2070 PRINT:PRINT
2080 PRINT "        Premi un tasto per uscir
e "
2090 GOSUB 1070
2100 GOTO 2120
2110 END
2120 REM
2130 REM ***********************
2140 REM *                     *
2150 REM *   SCELTA INIZIALE   *
2160 REM *                     *
2170 REM ***********************
2180 REM
2190 CLS
2200 LOCATE 11,3
2210 PRINT "DISASSEMBLER Z80"
2220 LOCATE 8,8
2230 PRINT "1. Lettura memoria"
2240 LOCATE 8,10
2250 PRINT "2. Memorizzazione dati"
2260 LOCATE 8,12
2270 PRINT "3. Fine programma"
2280 LOCATE 12,15
2290 PRINT "Premi un tasto"
2300 KK$=INKEY$
2310 IF KK$<"1" OR KK$>"3" THEN 2300
2320 ON VAL(KK$) GOTO 370,1800,2670
2330 END
2340 REM
2350 REM ***********************
2360 REM *                     *
2370 REM *    PRESENTAZIONE    *
2380 REM *                     *
2390 REM ***********************
2400 REM
2410 LOCATE 11,10
2420 PRINT "{5C219}  {3C219}   {3C219}"
2430 LOCATE 15,11
2440 PRINT "{C219} {C219}   {C219} {C219}
 {C219}"
2450 LOCATE 14,12
2460 PRINT "{C219}  {C219}   {C219} {C219}
 {C219}"
2470 LOCATE 13,13
2480 PRINT "{C219}    {3C219}  {C219}   {C
219}"
2490 LOCATE 12,14
2500 PRINT "{C219}    {C219}    {C219} {C21
9}  {C219}"
2510 LOCATE 11,15
2520 PRINT "{C219}     {C219}    {C219} {C2
19}  {C219}"
2530 LOCATE 11,16
2540 PRINT "{5C219}  {3C219}    {3C219}"
2550 RETURN
2560 END
2570 REM
2580 REM ***********************
2590 REM *                     *
2600 REM *       ERRORE        *
2610 REM *                     *
2620 REM ***********************
2630 REM
2640 IF ERL=2000 THEN PRINT "
    ERRORE":PRINT:BEEP:BEEP:RESUME 1980:GOT
O 1980
2650 IF ERL=1930 THEN PRINT "
    ERRORE":PRINT:BEEP:BEEP:RESUME 1920:GOT
O 1920
2660 END
2670 REM
2680 REM ***********************
2690 REM *                     *
2700 REM *   FINE PROGRAMMA    *
2710 REM *                     *
2720 REM ***********************
2730 REM
2740 CLS
2750 LOCATE 2,10
2760 BEEP:BEEP
2770 PRINT "SEI PROPRIO SICURO DI FINIRE <
S/N> "
2780 KK$=INKEY$
2790 IF KK$="" THEN 2780
2800 IF KK$="S" THEN CLS:KEY ON:END
2810 GOTO 2120
```

CORSO PRATICO DI LINGUAGGIO MACCHINA

di Piero Todorovich

Dopo aver illustrato il set di istruzioni della CPU Z80 ed effettuato i primi passi, proseguiamo il nostro discorso approfondendo la conoscenza dei registri del processore e della struttura di dati denominata stack.
I registri del processore sono le "variabili" disponibili nei programmi in LM.
Mentre in Basic possiamo utilizzare un numero molto elevato di variabili stringa/testo definendone nome, contenuto, e modo di utilizzo a nostro piacimento, in linguaggio macchina dobbiamo adattarci ad un numero di registri predeterminato, ciascuno capace di contenere solo numeri interi e per di più limitato nel campo di utilizzo.
I registri disponibili sono:
A, B, C, D, E, H, L, I, R registri semplici (possono contenere un byte cioè numeri interi compresi tra 0 e 255)
AF, BC, DE, HL, SP, IX, IY registri doppi (possono contenere due bytes cioè numeri interi compresi tra 0 e 65535)
I registri doppi (AF, BC, HL, DE) sono semplicemente l'associazione di due registri semplici pertanto non possono essere utilizzati in modo indipendente:
se assegnamo al registro doppio HL il valore 0, anche i registri semplici H e L saranno ovviamente azzerati.
Ad ogni registro corrisponde un certo numero di istruzioni che ne permette l'uso, da questo set dipende il campo di utilizzo del registro stesso.
Il registro semplice "A" detto "accumulatore" è quello più usato per ciò che riguarda le operazioni logiche e matematiche poichè ha il set più completo: nelle funzioni OR, AND e talvolta in altre, l'uso di A è addirittura sottinteso: troviamo "OR B" in luogo di un "OR A, B" ecc.
I registri semplici B, C, D, E, H, L sono utilizzabili in pari misura per caricare dati da utilizzare in funzioni logiche e matematiche con il registro A oppure per la maggioranza delle operazioni sui singoli bit (rotazioni, test, set, reset) e altre.
I registri "I" ed "R" hanno un uso riservato, rispettivamente: come vettore per routines interrupt e per il rinfresco (rigenerazione) della memoria, cose che per il momento non ci interessano.
I registri doppi hanno anch'essi un uso preferenziale: useremo HL per puntare locazioni di memoria (cioè come contenitore per la locazione), BC come contatore, DE per gli altri usi.
Il registro AF è utilizzato unicamente per memorizzare e richiamare all'occorrenza l'accumulatore (A) e il registro di flag (F) in una struttura di dati chiamata stack.
SP, (Stack pointer) punta alla locazione di memoria dove è sito lo stack.
I registri IX e IY hanno un uso specializzato, sono utilizzati normalmente come puntatori per buffer di dati di 256 bytes, grazie ad un set particolare di istruzioni che permette una facile gestione.
Per chi, arrivato a questo punto, ha già le idee confuse, chiariamo che, per un primo approccio con la programmazione, è sufficiente la conoscenza del sottoinsieme di istruzioni che riguarda i registri principali (A, D, E, B, C, HL, BC, DE) e dell'unica struttura di dati che utilizzeremo: lo Stack.
Lo stack è un'area della memoria che possiamo utilizzare per memorizzare temporaneamente il valore contenuto nei registri, per poi riutilizzarlo in altro momento.
Nella programmazione in linguaggio macchina, infatti, non possiamo permetterci il lusso di impegnare registri con dati non di immediato utilizzo; i comandi PUSH e POP seguiti dal nome di un registro doppio servono rispettivamente per memorizzare o richiamare un dato nello stack.
Lo stack ha una struttura a pila, tale che l'ultimo dato memorizzato è anche il primo ad essere richiamato, questa struttura è nota sotto il nome di LIFO (last input-first output). Lo stack è utilizzato oltre che dal programmatore, anche dalla CPU quando esegue chiamate a subroutine con il comando CALL (simile al GOSUB del Basic), per memorizzare la locazione a cui ritornare. Fate molta attenzione quindi che i vostri programmi LM non lascino dati nello stack all'esecuzione del comando RET (ritorno) altrimenti il dato letto dalla CPU come locazione di ritorno non sarà corretto e il risultato sarà un CRASH.
Il Crash è la perdita del controllo su ciò che la CPU sta facendo e capita frequentemente durante il collaudo di programmi contenenti errori. Infatti, mentre un programma Basic errato si blocca presentando un messaggio di errore, un programma LM con errori nell'uso dello stack o nelle istruzioni di "salto" prosegue spesso nella propria pazza corsa.
In tal caso la CPU interpreterà aree di dati, o valori casuali presenti in memoria come se si trattasse di istruzioni con risultati imprevedibili. Il più delle volte, per riprendere il controllo del proprio computer occorrerà resettare o spegnere e riaccendere la macchina.
Un registro particolare e di grande importanza nella programmazione è il registro di stato "F" (flag) che abbiamo volutamente lasciato per ultimo proprio per parlarne in modo approfondito.

A differenza degli altri registri semplici, F non dispone di nessuna istruzione che permetta di caricare un qualsiasi dato o eseguire operazioni, il suo uso è riservato all'indicazione di situazioni verificatesi in operazioni eseguite con gli altri registri.
Il confronto tra due numeri: CP n, tra il contenuto dell'accumulatore e il numero "n", non provoca altro che la modifica dello stato 1/0 di alcuni bit del registro di flag in dipendenza del risultato del confronto stesso.
Se l'accumulatore è minore del numero, il bit denominato Carry è posto a 1 altrimenti, nella condizione opposta (A>=n) il Carry vale 0.
Il bit Z del registro di Flag indica invece l'uguaglianza tra i due operandi se è settato a 1, altrimenti A<>n.
Per eseguire "decisioni" in un programma in linguaggio macchina occorre controllare i bit di flag con le istruzioni di salto o chiamata condizionata:
JP z,nnnn: salta alla locazione nnnn se z=1
Call nz,nnnn: salta alla subroutine della locazione nnnn se z=0 (infatti "nz" sta per NOT(z))
Si intuisce chiaramente come sia simile l'uso delle decisioni in Basic a quelle effettuate con il LM.
*) IF a<b THEN GOTO1000 (salta alla linea 1000 se la variabile a<b)
*) CP B
JP c,55000 (salta alla locazione 55000 se il registro A<B)

ARTICOLI

Un po' meno semplice cercare l'equivalenza di "a>b" o "a<=h" che vi lasciamo per esercizio.
Per dovere di cronaca, molte altre istruzioni oltre a "CP", mutano i bit di flag col verificarsi di particolari condizioni, inoltre esistono altri bit (oltre a "Z" e Carry), il cui valore si presta ad interessanti applicazioni.
Rimandiamo l'approfondimento prossimamente o per i più esigenti, su testi specifici.

PICCOLI TRUCCHI

ROUTINES INTERRUPT

Utilizzando il potente MSX BASIC vi sarete resi conto di come sia possibile utilizzare i comandi basic ON INTERVAL GOSUB, INTERVAL ON per la gestione di controlli o funzioni di sincronizzazione temporale di sprite in movimento ecc.
Programmando in linguaggio macchina è ugualmente possibile utilizzare l'interrupt, intercettando la chiamata effettuata dalla routine di ROM ogni 50' di secondo alla locazione FD9FH.
Sostituendo a tale locazione il codice 201 (RET) presente, con 205 (Call) e relativa locazione nei due bytes successivi, siamo in grado di inserire qualsiasi sottoprogramma LM self-made al normale interrupt.

LO SAPEVATE CHE...

L'evoluzione delle routine di lettura-scrittura da registratore controllate via software, ha raggiunto negli MSX un notevole livello di evoluzione. In particolare, la routine di Load si auto-sincronizza sul segnale da caricare anche se questo è accidentalmente o volutamente registrato a velocità differenti. Se il vostro registratore è di buona qualità, e consente la regolazione della velocità di scorrimento del nastro, potete caricare più speditamente qualsiasi programma già registrato semplicemente aumentando la velocità di scorrimento.

KEY PROGRAMMER

I tasti funzione MSX si prestano a tantissime applicazioni interessanti, in particolare oltre ai comandi Basic più usati è possibile inserire i caratteri di controllo più utili o associazioni tra questi:
Keyx, chr$(18)+chr$(32) per inserire spazi nei programmi
Keyx, chr$(18)+chr$(58) per inserire istruzioni basic
Keyx, chr$(21) cancella una linea su schermo
Keyx, chr$(14)+":rem" per aggiungere REMarks a fine linea

DIREZIONE, REDAZIONE E AMMINISTRAZIONE
Via Rosellini, 12 - 20124 Milano
Telefoni: (02) 68.03.68 - 68.00.54
68.80.951-2-3-4-5
Telex 333436 GEJ IT

SEDE LEGALE:
Via G. Pozzone, 55 - 20121 Milano

DIRETTORE RESPONSABILE:
Giampietro Zanga

COORDINAMENTO EDITORIALE:
Angelo Cattaneo - Piero Todorovich

GRAFICA E IMPAGINAZIONE:
Wilma Germani

FOTOCOMPOSIZIONE:
GDB fotocomposizione - Milano

STAMPA:
Grafika 78 - Pioltello - Milano

AUTORIZZAZIONE ALLA PUBBLICAZIONE:
Trib. di Milano n° 49 del 2-2-1987

Per la rivista non è prevista la sottoscrizione di abbonamenti

PUBBLICITÀ
Concessionario per l'Italia e l'Estero
J. Advertising s.r.l.
V.le Restelli, 5 - 20124 MILANO
Tel. (02) 68.82.895-68.80.606-68.87.233
Tlx 316213 REINA I

Concessionario esclusivo per la diffusione in Italia e Estero:
SODIP - Via Zuretti, 25
20125 MILANO

Spedizione in abbonamento postale Gruppo III/70
Prezzo della rivista L. 8.000
Numeri arretrati L. 16.000